Guida alla conversione da Simatic S7 a Logix5000

Soluzione per le applicazioni

Allen-Bradley • Rockwell Software

Rockwell Automation

Informazioni importanti per l'utente

Un dispositivo a stato solido ha caratteristiche di funzionamento differenti da quelle di un dispositivo elettromeccanico. Safety Guidelines for the Application, Installation and Maintenance of Solid State Controls (pubblicazione SGI-1.1 disponibile presso il proprio rivenditore locale Rockwell Automation oppure online all'indirizzo http://literature.rockwellautomation.com) descrive alcune importanti differenze tra i dispositivi a stato solido e i dispositivi elettromeccanici cablati. A seguito di tali differenze e della grande varietà di utilizzo dei dispositivi a stato solido, tutte le persone responsabili dell'applicazione di questi dispositivi devono verificare che qualsiasi applicazione prevista per questa apparecchiatura sia accettabile.

In nessun caso Rockwell Automation, Inc. sarà responsabile o perseguibile per danni indiretti o derivanti dall'utilizzo o dall'applicazione di questo dispositivo.

Gli esempi e gli schemi in questo manuale sono inclusi esclusivamente a scopo illustrativo. A causa delle molteplici variabili ed esigenze associate ad ogni specifica installazione, Rockwell Automation, Inc. non si assume la responsabilità e non è perseguibile per l'utilizzo effettivo basato su esempi e schemi.

Rockwell Automation, Inc. non si assume alcuna responsabilità sui brevetti in relazione all'uso di informazioni, circuiti, dispositivi o software descritti in questo manuale.

È vietata la riproduzione, intera o parziale, dei contenuti di questo manuale senza il consenso scritto di Rockwell Automation, Inc.

In questo manuale sono utilizzate delle note per avvertire l'utente sulle considerazioni relative alla sicurezza.

AVVERTENZA	Identifica le informazioni su procedure o circostanze che possono causare esplosioni in ambienti pericolosi, provocando lesioni personali o decesso, danni alle proprietà o perdite economiche.
IMPORTANTE	Identifica le informazioni critiche per un buon funzionamento dell'applicazione e la comprensione del prodotto.
ATTENZIONE	Identifica le informazioni su procedure o circostanze che possono causare lesioni personali o decesso, danni alle proprietà o perdite economiche. Gli avvisi aiutano l'utente a identificare un pericolo, evitare un pericolo e riconoscerne le conseguenze.
RISCHIO DI FOLGORAZIONI	Eventuali etichette sopra o all'interno dell'apparecchiatura, ad esempio un inverter o un motore, avvertono gli utenti che può essere presente una tensione pericolosa.
RISCHIO DI USTIONI	Eventuali etichette sopra o all'interno dell'apparecchiatura, ad esempio un inverter o un motore, avvertono gli utenti che le superfici possono raggiungere temperature pericolose.

Allen-Bradley, Rockwell Automation e TechConnect sono marchi commerciali di Rockwell Automation, Inc.

I marchi commerciali che non appartengono a Rockwell Automation sono proprietà delle rispettive società.

Sommario

Prefazione

Scopo 7
Differenza tra conversione e traduzione 7
Terminologia 8
Risorse aggiuntive 8
Servizi di conversione della logica dei PLC di Rockwell Automation ... 9
- Caratteristiche dei servizi 9
- Servizi di conversione programmi PLC “one-stop” 9
- Vantaggi dei servizi 10
- Servizi offerti 10
- Pacchetto di conversione di base 10
- Pacchetto di conversione e clean-up iniziale 10
- Opzioni aggiuntive 11
- Altri pacchetti di conversione disponibili 11

Capitolo 1

Conversione hardware

Introduzione 13
Controllori S7 13
Sistemi I/O 14
- I/O locale di S7 14
- Selezione e configurazione dei componenti I/O per S7 14
- I/O locale di Logix 16
- Scelta e configurazione dei componenti I/O di Logix 18
- I/O remoto di S7 20
- Configurazione dell'I/O remoto di S7 su Profibus DP 21
- I/O distribuito Logix 22
- Configurazione dell'I/O distribuito Logix 22

Reti 25
- Reti in S7 25
- Reti in Logix 27

Conversione dell'interfaccia operatore 31
Conversione dei sistemi contenenti controllori distribuiti 32
- Implementazione di hardware e software 32

Collegamento di dispositivi Siemens e Rockwell Automation 34
- Controllori 34
- Dispositivi distribuiti 34

Capitolo 2

Funzioni di Logix che potrebbero non essere familiari agli utenti di S7

Introduzione 35
Confronto tra blocchi organizzativi (OB) di S7 e task Logix 36
- Blocchi organizzativi in S7 36
- Task in Logix 41
- Monitoraggio task 46

Tag senza indirizzi 47
- Aree di dati in S7 47
- Dati in Logix 50

I/O e tag alias 51
Linguaggi di programmazione 53

Linguaggio ladder Logix 54
Testo strutturato Logix 54
Diagramma a blocchi funzione Logix 55
Diagramma funzionale sequenziale Logix 55
Conversione del codice STEP 7 in Logix 55
Array non puntatori 56
Istruzioni add-on 57
Panoramica delle istruzioni add-on 57
Tag di supporto 58
Common Industrial Protocol (CIP) 58
Visualizzazione della rete 59
Scambio di dati tra controllori 60
Invio/ricezione in STEP 7 60
Tag prodotti/consumati in Logix 60
Tipi di dati definiti dall'utente 61
Aggiornamento asincrono degli I/O 62
Tipo di dati DINT 62
Phase Manager 63
Gestione delle fasi in STEP 7 63
PhaseManager in Logix 63
Tempo di sistema coordinato (CST) 65
Ingressi con registrazione cronologica 65
Uscite schedulate 65
Nessuna variabile temporanea 66
Non sono necessari accumulatori o registri speciali 66

Capitolo 3

Conversione del software di sistema e delle funzioni standard

Introduzione 67
Funzioni del sistema Logix 68
Copia 68
Impostazione o lettura di data e ora 69
Lettura dell'ora del sistema 69
Gestione degli interrupt 70
Errori 70
Stato – Controllore 71
Stato – Modulo 71
Stato – per OB e task 72
Timer 72
Routine di conversione 73
Routine di gestione delle stringhe 73
Esempi di chiamate di funzione al sistema 74
Impostazione dell'orologio 74
Disabilitazione degli interrupt 76

Lettura dell'ora del sistema 78
Determinazione degli errori 79
Informazioni sul modulo 80
Determinazione del tempo di scansione 81

Capitolo 4

Conversione delle strutture di programma tipiche

Introduzione 83
Esempi del codice di conversione 83
Traduzione della logica ladder 83
Salti e processo decisionale 90
Array 94
Tipi di dati utente 99
Puntatori e array 102
Macchina a stati 103
Macchina a stati di STEP 7 104
Stringhe 108
Variabili temporanee di STEP 7 110
Funzioni 110
Copia di blocco, COP e CPS 114
Espressioni matematiche 116
Altri argomenti relativi alla programmazione 120
Ambito delle variabili 120
OB, task e schedulazione 120
Un esempio più ampio - Modulo di controllo 121
Componenti di CM 121
Valvola con tipo di dati utente 122
Istruzione add-on 123
Dati locali dell'istruzione add-on 124
Richiamo 127

Capitolo 5

Errori comuni durante la conversione in Logix

Introduzione 129
Se non si seleziona l'hardware appropriato 129
Sottovalutazione dell'impatto della schedulazione dei task 130
Esecuzione della traduzione anziché della conversione 130
Se non si utilizzano i linguaggi appropriati 130
Implementazione di tipi di dati errati – DINT e INT 131
Somma di DINT 131
Somma di INT 131
Risultati temporali 131
Codice utente che emula istruzioni esistenti 132
Codice utente 132
Istruzione COP 132
Utilizzo errato di COP, MOV e CPS 133
Utilizzo errato di CPT 133

Se le stringhe non sono gestite in modo ottimale 133
Utilizzo estensivo di salti 133
Se non si utilizzano tag alias 133

Capitolo 6

Glossario S7 - Logix

Introduzione 135
Terminologia hardware 135
Terminologia software 136

Appendice A

Componenti di S7 300 e S7 400 ed equivalenti RA

Introduzione 139
CPU Compact S7 300 140
CPU S7 300 standard 140
CPU Technology S7 300 141
CPU S7 300 fail-safe 142
Moduli di ingresso digitali S7 300 142
Moduli di uscita digitali S7 300 143
Moduli di uscita a relè S7 300 144
Moduli digitali misti S7 300 144
Moduli di ingresso analogici S7 300 144
Moduli di uscita analogici S7 300 145
Moduli analogici misti S7 300 146
Controllori S7 400 standard 146
Controllori ridondanti e fail-safe 147
Moduli di ingresso digitali 147
Moduli di uscita digitali 147
Moduli di ingresso analogici 148
Moduli di uscita analogici 148

Appendice B

Tabella di riferimento incrociato dei pannelli operatore Siemens

Pannelli SIMATIC Micro ed equivalenti Rockwell Automation 149
Pannelli SIMATIC - serie 7x ed equivalenti Rockwell Automation 151
Pannelli SIMATIC - serie 17x ed equivalenti Rockwell Automation 152
Pannelli SIMATIC - serie 27x ed equivalenti Rockwell Automation 154
Pannelli SIMATIC Multi Panel - serie 27x ed equivalenti Rockwell Automation 156
Pannelli SIMATIC Multi Panel - serie 37x ed equivalenti Rockwell Automation 158
Assistenza Rockwell Automation 162
Assistenza per l'installazione 162
Restituzione di prodotti nuovi non funzionanti 162

Scopo

Il presente manuale rappresenta una guida per gli utenti e i tecnici che hanno utilizzato i sistemi di controllo basati su una delle due seguenti piattaforme:

- controllore Siemens S7
- PAC (Programmable Automation Controller, controllore di automazione programmabile) Logix di Rockwell Automation

E inoltre

- desiderano utilizzare le funzioni dei PAC o sono nelle fasi iniziali del processo di migrazione da S7 a Logix.
- dispongono di un codice di programma STEP 7 che desiderano convertire in codice per RSLogix 5000.

Utilizzare questo manuale per apprendere le procedure opportune ed evitare errori durante la conversione del progetto in Logix.

Differenza tra conversione e traduzione

La differenza tra conversione e traduzione viene affrontato ripetutamente in questa guida alla conversione delle applicazioni. La traduzione si concentra solo sulla riga di codice e sulla ricerca di un equivalente nei linguaggi Logix. Per convertire un'applicazione in modo ottimale, è necessario andare oltre la traduzione. Ad esempio, è possibile trarre vantaggio dalla scelta di un linguaggio di programmazione diverso, dall'utilizzo di varie tecniche di programmazione e dalla progettazione di un altro schema di schedulazione per risolvere lo stesso task. Pertanto, la conversione viene eseguita in un contesto di progettazione e conoscenza dei punti di forza del sistema Logix di livello più alto.

Se è necessario convertire codice applicativo, occorre conoscere il programma STEP 7 prima di avviare la conversione, o perchè direttamente coinvolti nello sviluppo del programma oppure leggendo la documentazione del programma e dei processi che controlla. Se il programma o il processo non è conosciuto o documentato a sufficienza, sarà difficile eseguire la conversione in modo corretto; verrà eseguita la semplice traduzione ed è probabile che l'operazione non riesca. Ad esempio, in Logix, esiste uno spazio nomi globale, mentre nell'ambiente Siemens sono presenti blocchi dati che possono essere caricati/scaricati tramite il codice applicativo. La valutazione di questi fattori consente di studiare una strategia per la conversione.

In alcuni casi, se la documentazione del processo e del programma non è sufficiente, può risultare più efficace in termini di durata/costo del progetto complessivo redigere nuove specifiche e avviare il programma Logix con tempo minimo dedicato alla traduzione dal programma precedente.

Terminologia

STEP 7 è un software di programmazione per i controllori SIMATIC S7 di Siemens. Il software RSLogix 5000 è utilizzato con i controllori di automazione programmabili Logix di Rockwell Automation. Logix viene definito controllore di automazione programmabile perché le sue funzioni sono molto più estese rispetto a un PLC tradizionale generico. Fornisce una piattaforma di controllo eccellente per un controllo multidisciplinare, uno spazio nomi comune, tempo di sistema coordinato per architetture multi-CPU realmente scalabili, tipi di dati definiti dall'utente e connettività NetLinx completa.

Il termine “Logix” è utilizzato in riferimento a uno qualsiasi dei controllori ControlLogix, CompactLogix, GuardLogix, FlexLogix, DriveLogix o SoftLogix o all'ambiente di programmazione RSLogix 5000, e risulta chiaro dal contesto a quale elemento si fa riferimento.

Risorse aggiuntive

In ogni sezione di questa guida viene fatto riferimento ad altri manuali dell'utente, guide alla scelta e documenti di Rockwell Automation in cui è possibile consultare ulteriori informazioni.

Numero pubblicazione	Titolo pubblicazione
1756-SG001	ControlLogix - Guida alla scelta
1769-SG001	1769 CompactLogix - Guida alla scelta
1768-UM001	1768 CompactLogix Controllers User Manual
1769-SG002	Compact I/O Selection Guide
1756-RM094	Considerazioni sulla progettazione dei controllori Logix5000 - Manuale di Riferimento
1756-PM001	Logix5000 Controllers Common Procedures Programming Manual
1756-RM003	Logix5000 Controllers General Instructions Reference Manual
1734-SG001	POINT I/O Selection Guide
1738-SG001	ArmorPoint I/O Selection Guide
1792-SG001	ArmorBlock MaXum I/O and ArmorBlock I/O Selection Guide
1794-SG002	FLEX I/O and FLEX Ex Selection Guide
NETS-SG001	NetLinx Selection Guide
VIEW-SG001	Visualization Platforms Selection Guide
IA-RM001	Integrated Architecture: Foundations of Modular Programming
6873-SG004	Encompass Program Product Directory
1756-PM010	Logix5000 Controllers Add-On Instructions Programming Manual
1756-RM087	Logix5000 Controllers Execution Time and Memory Use Reference Manual
IASIMP-RM001	IA Recommended Literature Reference Manual

Servizi di conversione della logica dei PLC di Rockwell Automation

Rockwell Automation fornisce servizi aggiuntivi per la conversione della logica dei PLC.

- Caratteristiche dei servizi
- Servizi di conversione programmi PLC “one-stop”
- Vantaggi dei servizi
- Servizi offerti
- Pacchetto di conversione di base
- Pacchetto di conversione e clean-up iniziale
- Altri pacchetti di conversione disponibili

Caratteristiche dei servizi

I servizi di conversione dei programmi consentono di convertire il programma di un precedente PLC Allen-Bradley o di un controllore programmabile di terze parti per essere eseguito su un sistema di controllo programmabile Logix o sui controllori programmabili SLC 500/MicroLogix o PLC-5.

I prodotti precedenti sono spesso costosi da gestire e difficili da riparare, il che può fare aumentare i tempi di fermata e ridurre la produzione. Per questo motivo, l'Assistenza Clienti Rockwell Automation ora offre i servizi di conversione dei programmi. Questi servizi sono studiati per ridurre i costi e il tempo necessario per migrare da un vecchio PLC a una delle nostre nuove piattaforme di controllo PAC o PLC.

Servizi di conversione programmi PLC “one-stop”

La migrazione ad una nuova piattaforma di controllo Allen-Bradley da un prodotto precedente migliorerà il processo di produzione, l'affidabilità e la flessibilità del sistema, garantirà una migliore potenza di calcolo e ridurrà i costi di riparazione delle apparecchiature e il magazzino ricambi. Con i servizi di conversione programmi dell'Assistenza Clienti di Rockwell Automation, il programma del vecchio controllore programmabile verrà convertito in modo facile e veloce nella nuova famiglia di controllori. I tecnici dell'Assistenza Clienti sono in grado di fornire assistenza per la migrazione di apparecchiature Allen-Bradley precedenti o per convertire i sistemi PLC nei prodotti Rockwell Automation, riducendo al minimo i tempi di fermo e ottimizzando i risultati operativi.

Vantaggi dei servizi

Nel processo di conversione dei programmi vengono impegnati specialisti di ciascuna piattaforma di prodotti. Non è difficile rilevare anomalie nella logica causate da errori di digitazione. Nella maggior parte dei casi, viene riprodotta l'intera tabella dati e non vi saranno perdite di dati, la documentazione originale verrà conservata e non sarà necessario digitare nuovamente commenti e simboli. I programmi originali Allen-Bradley possono essere in formato 6200, APS o serie AI. I nuovi programmi avranno il formato RSLogix appropriato.

Servizi offerti

Sono disponibili due pacchetti di conversione programmi nonché pacchetti personalizzati specifici a seconda delle necessità.

Pacchetto di conversione di base

- Il programma del controllore programmabile originale viene convertito nel formato ControlLogix, CompactLogix, PLC-5 o SLC 500/MicroLogix appropriato.
- Il pacchetto fornisce un elenco degli errori generato durante la conversione che include istruzioni non convertibili direttamente ed eventuali indirizzi non convertiti, che potrebbero includere puntatori e indirizzamento indiretto.
- Il programma e l'elenco degli errori viene restituito al cliente per il debug e la correzione manuali.

Pacchetto di conversione e clean-up iniziale

- Il programma del controllore programmabile originale viene convertito nel formato ControlLogix, PLC-5 o SLC 500/MicroLogix appropriato.
- Eventuali errori di istruzioni e/o indirizzamento saranno corretti nella nuova famiglia di processori.
- Il programma completo sarà quindi restituito al cliente per l'avvio e il debug finali.

Opzioni aggiuntive

Di seguito sono riportate le opzioni aggiuntive ai pacchetti:

- assistenza telefonica durante la fase di avviamento e debug del progetto
- consulenza su reingegnerizzazione del sistema, interfaccia operatore, architettura e strategie di comunicazione. Per sfruttare al massimo le capacità di controllo della nuova piattaforma, non facenti parte di un processo di traduzione del codice, sono disponibili servizi aggiuntivi di addestramento e avvio on-site. Informazioni presso l'ufficio Global Sales and Solutions (GSS) di zona
- migrazione completa "chiavi in mano" o aggiornamenti disponibili presso l'ufficio GSS/Engineered Systems di zona.

Altri pacchetti di conversione disponibili

- Dal formato PLC-2 al formato ControlLogix, CompactLogix, PLC-5, SLC500/MicroLogix
- Dal formato PLC-3 al formato ControlLogix, CompactLogix o PLC-5
- Dal formato PLC-5/250 al formato ControlLogix o CompactLogix
- Da Modicon – Quantum, 984, 584, 380, 381, 480, 485, 780, 785 al formato ControlLogix o CompactLogix
- Da Siemens – S-5, S-7 al formato ControlLogix o CompactLogix
- Da TI - 520, 520C, 525, 530, 530C, 535, 560, 560/565, 565, 560/560T, 560T, 545, 555, 575 al formato ControlLogix o CompactLogix
- Da GE serie 6 al formato ControlLogix o CompactLogix

Sono inoltre disponibili conversioni di programmi di altri controllori programmabili di terze parti in formato per controllori Allen-Bradley. Per ulteriori informazioni, rivolgersi all'Assistenza Tecnica.

Per pianifiare un progetto di conversione o ricevere ulteriori informazioni sui servizi di conversione dei programmi, rivolgersi all'ufficio vendite di Rockwell Automation di zona o a un distributore autorizzato. È possibile inviarci un messaggio e-mail all'indirizzo raprogramconversion@ra.rockwell.com o visitare il sito http://support.rockwellautomation.com/ e visualizzare il documento della KnowledgeBase G19154.

IMPORTANTE Utilizzare i servizi di consulenza per la reingegnerizzazione per espandere le funzionalità di sistema e non per modificare l'hardware perché obsoleto o per motivi correlati.
Le conversioni dal formato SLC al formato Logix e da PLC-5 al formato Logix e la generazione di commenti PCE sono integrate nel software RSLogix 5000.

Note:

Conversione hardware

Introduzione

L'obiettivo di questo capitolo è fornire una guida a utenti o tecnici per determinare l'hardware Logix corretto per sostituire l'apparecchiatura S7 esistente.

Nel capitolo viene illustrato come selezionare controllori, I/O locale, I/O remoto, reti e interfaccia operatore. È inoltre inclusa una sezione relativa all'architettura dei controllori distribuiti e sono forniti esempi di conversione hardware dei moduli S7 utilizzati più frequentemente.

Argomento	Pagina
Controllori S7	13
Sistemi I/O	14
Reti	25
Conversione dell'interfaccia operatore	31
Conversione dei sistemi contenenti controllori distribuiti	32
Collegamento di dispositivi Siemens e Rockwell Automation	34

Controllori S7

Nella seguente tabella è riportata una selezione dei principali controllori Siemens S7 correnti, utilizzati per un'ampia gamma di applicazioni.

Selezione di controllori Siemens S7 correnti

Controllore	Codice prodotto	Equivalente Logix
313C	6ES7 313-5BF03-0AB0	L23 seriale
314C-DP	6ES7 314-6CG03-0AB0	L23 EtherNet/IP, L31
315-2 DP	6ES7 315-2AG10-0AB0	L32E, L32C
317-2 DP	6ES7 317-6TJ10-0AB0	L35CR, L35E
317T-2 DP	6ES7 317-6TJ10-0AB0	L43, L45
319-3 PN/DP	6ES7 318-3EL00-0AB0	L45, L61
414-2	6ES7 414-2XK05-0AB0	L61, L62
414-3 414-3 PN/DP	6ES7 414-3XM05-0AB0 6ES7 414-3EM05-0AB0	L62, L63, L64, L65

Selezione di controllori Siemens S7 correnti

315F-2 PN/DP (sicurezza)	6ES7 315-2FH13-0AB0 6ES7 317-2FK13-0AB0	GuardLogix L61S, L62S, L63S
414-H (ridondante) 417-H	6ES7 414-4HM14-0AB0 6ES7 417-4HT14-0AB0	L61-L65 con SRM
PCS7 – Utilizza il controllore 417-4		L3x, L4x, L6x + FactoryTalk View, software FactoryTalk Batch

Di seguito è riportata una guida all'idoneità di alcuni dei controllori S7 utilizzati più frequentemente:

- S7 315-2DP: macchine di piccole e medie dimensioni.
- S7 317-2DP: macchine di dimensioni medie e medio-grandi, applicazioni di dimensioni piccole e medie per il controllo di processo.
- S7 414-2: controllo macchine complesse, applicazioni per il controllo di processo.
- S7 414-3: controllo macchine complesse, applicazioni per il controllo di processo di grandi dimensioni.

La gamma completa di controllori S7 è elencata nell'Appendice A.

Sistemi I/O

In queste sezioni sono descritti i sistemi I/O di Logix da utilizzare in sostituzione dell'apparecchiatura S7 esistente.

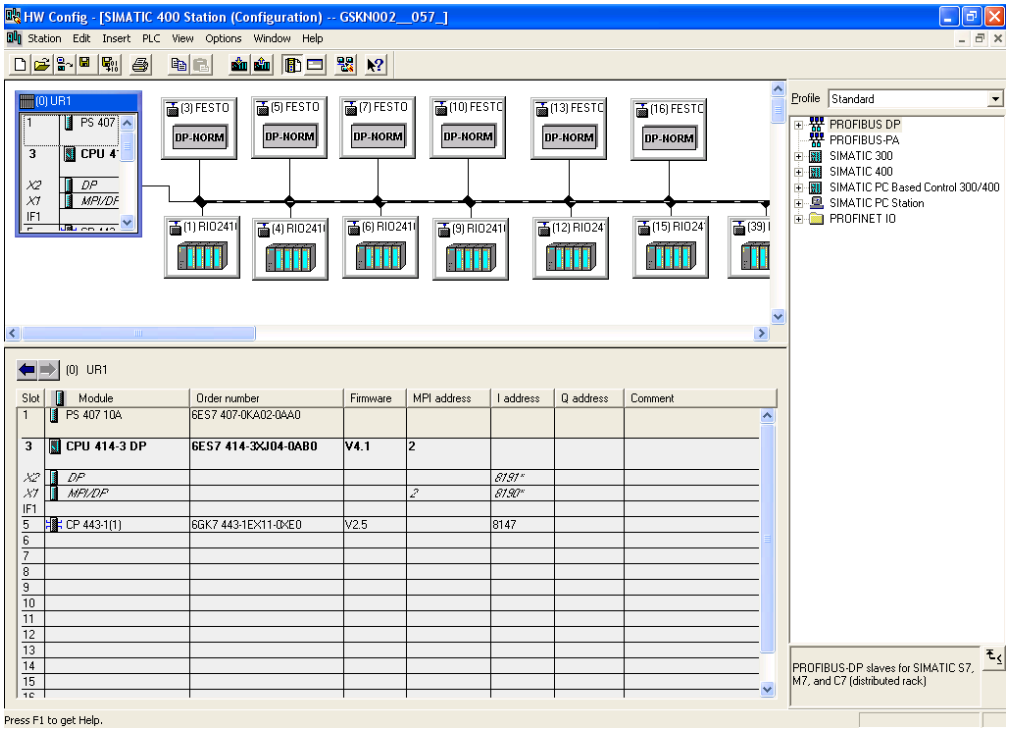
I/O locale di S7

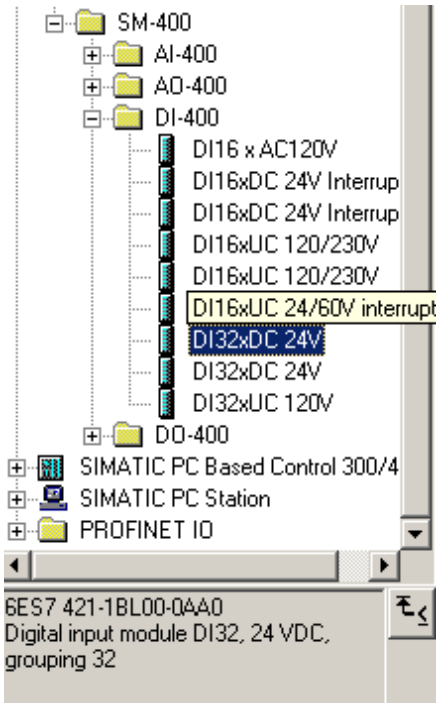
È disponibile una vasta gamma di moduli I/O per S7-300 e S7-400. I moduli per S7-300 sono montati su guida DIN standard e collegati alle schede adiacenti utilizzando connettori a U, forniti con i moduli. I moduli per S7-400 sono montati nel rack dell'S7-400.

Selezione e configurazione dei componenti I/O per S7

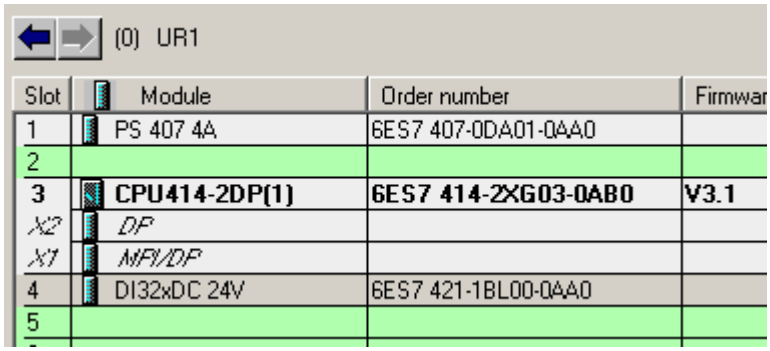
Le immagini riportate di seguito rappresentano il programma di configurazione hardware STEP 7, un programma separato nel gruppo di applicazioni STEP 7. Nel software RSLogix 5000 questa funzionalità è completamente integrata e verrà descritta in dettaglio più avanti nel presente manuale dell'utente.

Programma di configurazione hardware STEP 7





Trascinare il modulo selezionato nella schermata di configurazione del rack.



I/O locale di Logix

È disponibile una vasta gamma di moduli I/O per ControlLogix e CompactLogix. L'I/O 1769 ha dei ~~consti~~ contenuti e include solo le funzionalità necessarie, come spesso richiedono gli OEM, mentre la famiglia di I/O 1756 garantisce caratteristiche/funzioni avanzate per soddisfare le applicazioni più esigenti, come spesso viene richiesto dagli utilizzatori finali e per soddisfare livelli di rendimento specifici.

I moduli CompactLogix sono montati su guida DIN standard e un sistema di accoppiamento speciale fissa i collegamenti meccanici ed elettrici ai moduli adiacenti. I tecnici apprezzeranno il sistema di accoppiamento meccanico: con l'S7-300, i moduli sono fissati solo a una guida speciale e non tra di loro (a parte il connettore a U elettrico).

I moduli ControlLogix sono montati nei rack 1756.

- Per i controllori 1769-L31, 1769-L32C, 1769-L32E e 1768-L43, il numero massimo di moduli I/O collegati al rack del controllore è 16 in un massimo di 3 gruppi.
- Per i controllori 1769-L35CR, 1769-L35E e 1768-L45, il numero massimo di moduli I/O collegati al rack del controllore è 30, anch'essi suddivisi in 3 gruppi.
- Per i controllori 1756, il numero di slot nel rack definisce il numero massimo di moduli I/O locali, che può essere 4, 7, 10, 13 o 17.

Su entrambe le piattaforme è possibile collegare in rete I/O aggiuntivi tramite le reti CIP, in cui EtherNet/IP e ControlNet assicurano una perfetta integrazione degli I/O.

Nella seguente tabella sono elencati gli equivalenti Logix di alcuni diffusi moduli I/O S7.

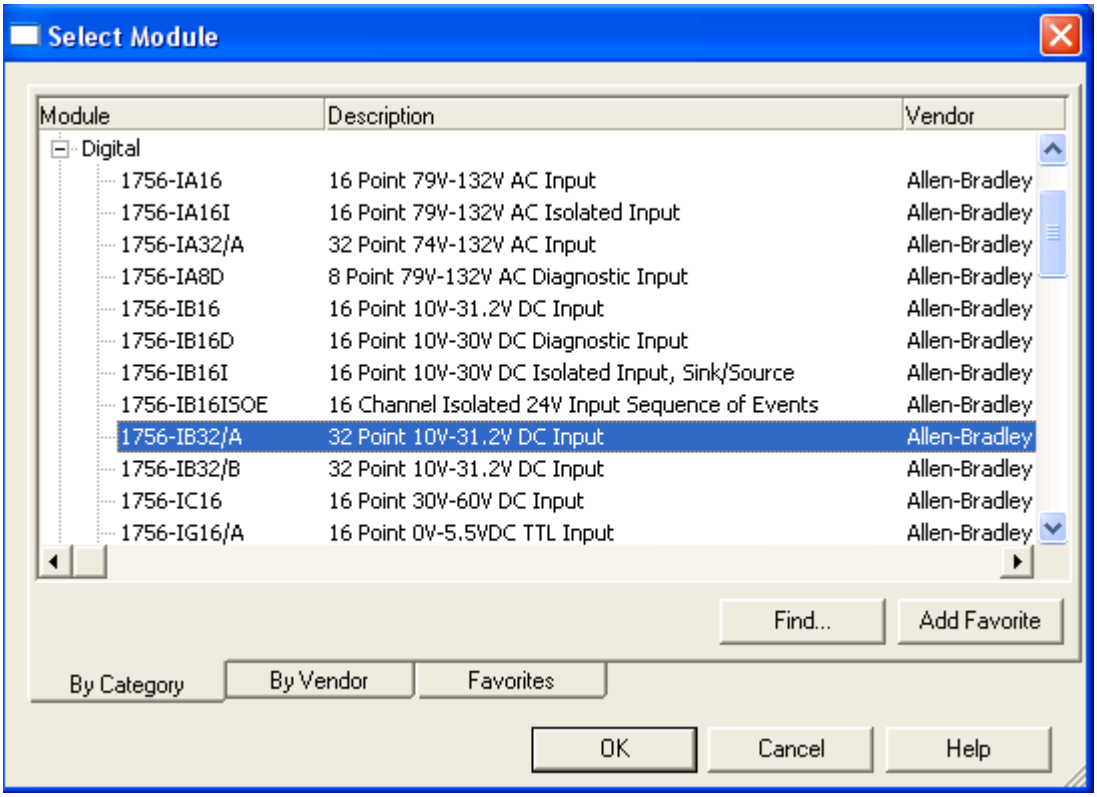
Equivalenti Logix dei moduli I/O ~~di~~ S7

Modulo I/O S7	Descrizione	Equivalente Logix	Descrizione
6ES7 321-1BL00-0AA0	Ingresso digitale a 32 canali S7-300	1769-IQ32	Ingresso digitale a 32 canali CompactLogix
6ES7 322 - 1BH01-0AA0	Uscita digitale a 16 canali S7-300	1769-OB16	Uscita digitale a 16 canali CompactLogix
6ES7 421-1BL01-0AA0	Ingresso digitale a 32 canali S7-400	1756-IB32	Ingresso digitale a 32 canali ControlLogix
6ES7 422-1BH01-0AA0	Uscita digitale a 16 canali S7-400	1756-OB16E	Uscita digitale a 16 canali ControlLogix

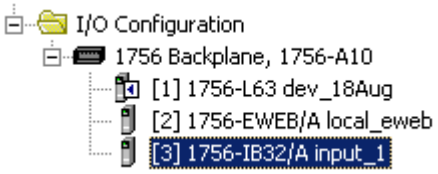
Per tabelle di conversione dei moduli I/O più dettagliate, fare riferimento all'Appendice A.

Scelta e configurazione dei componenti I/O di Logix

Dalla configurazione I/O nella struttura ad albero del progetto, è possibile accedere alla libreria Logix dei profili dei dispositivi. Tali profili consentono una configurazione completa e facile della tabella dati tramite procedure guidate oltre che il controllo programmabile ed intuitivo delle funzionalità di ciascun modulo, come conversione in scala, generazione di allarmi e diagnostica.


Select Module
Module
Description
Vendor
Digital
1756-IA16 16 Point 79V-132V AC Input Allen-Bradley
1756-IA16I 16 Point 79V-132V AC Isolated Input Allen-Bradley
1756-IA32/A 32 Point 74V-132V AC Input Allen-Bradley
1756-IA8D 8 Point 79V-132V AC Diagnostic Input Allen-Bradley
1756-IB16 16 Point 10V-31.2V DC Input Allen-Bradley
1756-IB16D 16 Point 10V-30V DC Diagnostic Input Allen-Bradley
1756-IB16I 16 Point 10V-30V DC Isolated Input, Sink/Source Allen-Bradley
1756-IB16ISOE 16 Channel Isolated 24V Input Sequence of Events Allen-Bradley
1756-IB32/A 32 Point 10V-31.2V DC Input Allen-Bradley
1756-IB32/B 32 Point 10V-31.2V DC Input Allen-Bradley
1756-IC16 16 Point 30V-60V DC Input Allen-Bradley
1756-IG16/A 16 Point 0V-5.5VDC TTL Input Allen-Bradley
Find...
Add Favorite
By Category
By Vendor
Favorites
OK
Cancel
Help

Selezionare un elemento per visualizzarlo nel rack della configurazione I/O.



I tag di profilo del dispositivo per il nuovo modulo I/O sono stati aggiunti automaticamente al database dei tag del controllore.

Nella seguente vista sono mostrati i tag parzialmente espansi.

⊟ Local:3:C			AB:1756_DI:C:0
⊞ Local:3:C.FilterOffOn_0_7			SINT
⊞ Local:3:C.FilterOnOff_0_7			SINT
⊞ Local:3:C.FilterOffOn_8_15			SINT
⊞ Local:3:C.FilterOnOff_8_15			SINT
⊞ Local:3:C.FilterOffOn_16_23			SINT
⊞ Local:3:C.FilterOnOff_16_23			SINT
⊞ Local:3:C.FilterOffOn_24_31			SINT
⊞ Local:3:C.FilterOnOff_24_31			SINT
⊞ Local:3:C.COSOnOffEn			DINT
⊞ Local:3:C.COSOffOnEn			DINT
⊞ Local:3:I			AB:1756_DI:I:0

Il profilo contiene dati di configurazione e di stato nonché dati I/O.

⊟ Local:0:C			AB:1756_DI:C:0
⊞ Local:0:C.FilterOffOn_0_7			SINT
⊞ Local:0:C.FilterOnOff_0_7			SINT
⊞ Local:0:C.FilterOffOn_8_15			SINT
⊞ Local:0:C.FilterOnOff_8_15			SINT
⊞ Local:0:C.FilterOffOn_16_23			SINT
⊞ Local:0:C.FilterOnOff_16_23			SINT
⊞ Local:0:C.FilterOffOn_24_31			SINT
⊞ Local:0:C.FilterOnOff_24_31			SINT
⊞ Local:0:C.COSOnOffEn			DINT
⊞ Local:0:C.COSOffOnEn			DINT
⊟ Local:0:I			AB:1756_DI:I:0
⊞ Local:0:I.Fault			DINT
⊞ Local:0:I.Data			DINT

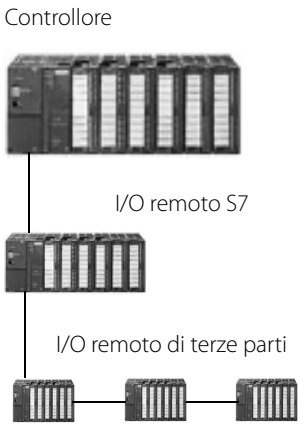
Per ulteriori informazioni, fare riferimento al Capitolo 4.

I/O remoto di S7

L'I/O viene solitamente diviso tra il rack locale del controllore e le stazioni dell'I/O remoto con la comunicazione eseguita sulla rete Profibus DP. Di seguito sono riportati i tipi di nodo Profibus DP:

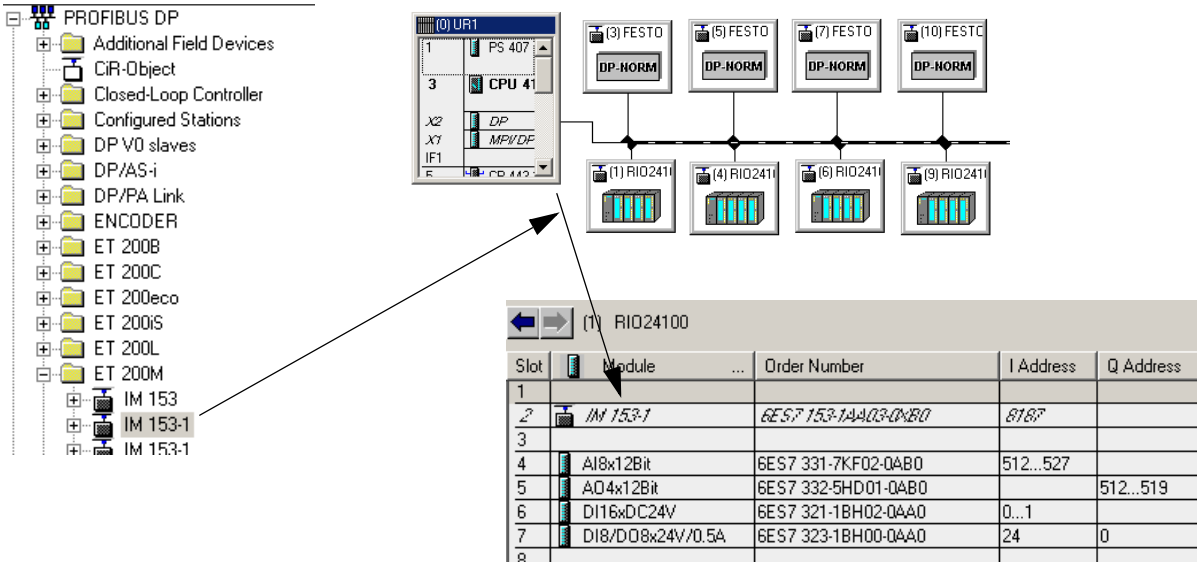
- I/O remoto S7, per cui i moduli I/O S7-300 standard sono montati in un pannello I/O remoto e si interfacciano con il bus Profibus DP tramite un modulo speciale. Il controllore rileva tale I/O come I/O locale e assegna gli indirizzi I/O standard. È denominato ET200M.
- Altri I/O remoti Siemens, come ET200S (simile al sistema POINT I/O) e ET200L (simile al sistema FLEX I/O).
- I/O remoto di terze parti. Alcuni produttori di I/O e valvole producono un'interfaccia per collegare i propri sistemi al bus Profibus DP allo stesso modo dell'I/O remoto S7. Per tali sistemi, può essere necessario importare un file di integrazione speciale (file GSD) in STEP 7.
- Alcuni produttori di dispositivi più complessi, come bilance e inverter a velocità variabile (VSD), producono interfacce Profibus DP per i propri prodotti. Per tali sistemi, sarà necessario importare un file di integrazione speciale (file GSD) in STEP 7. Per informazioni sul significato delle aree di dati, è spesso necessario fare riferimento alla documentazione fornita dal produttore.

Configurazione I/O tipica di S7



Configurazione dell'I/O remoto di S7 su Profibus DP

È possibile installare un modulo di interfaccia Profibus DP nella configurazione hardware eseguendo il trascinamento dal catalogo hardware sul grafico del bus Profibus DP. Una volta installato, il modulo di interfaccia può essere aperto ed è possibile aggiungere moduli S7-300 standard come se fosse l'I/O locale.



La tabella dati definisce gli indirizzi I/O associati all'inverter. I simboli relativi a tali indirizzi vengono aggiunti manualmente nella tabella dei simboli.
La configurazione hardware è stata completata.

È possibile utilizzare i dispositivi remoti sulla rete Profibus DP insieme a Logix, con gli stessi limiti/limitazioni di utilizzo presenti nell'ambiente S7.

I/O distribuito Logix

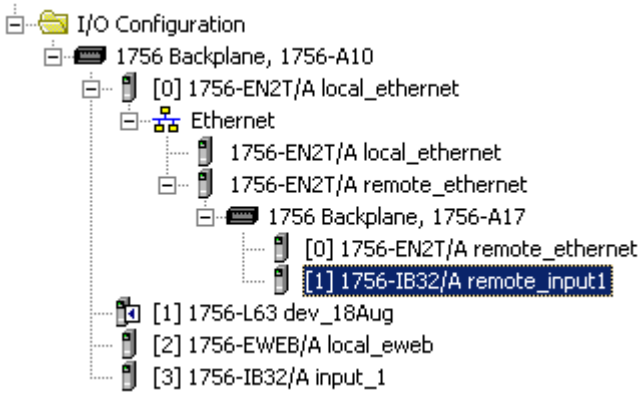
L'I/O distribuito di Rockwell Automation include l'I/O remoto che utilizza i moduli I/O 1756 o 1769 e diverse piattaforme di I/O distribuiti, come i sistemi POINT I/O, FLEX I/O, ArmorPoint e ArmorBlock.

I moduli I/O sono collegati alla rete tramite un modulo di comunicazione o una scheda di comunicazione oppure direttamente tramite un'interfaccia di comunicazione incorporata.

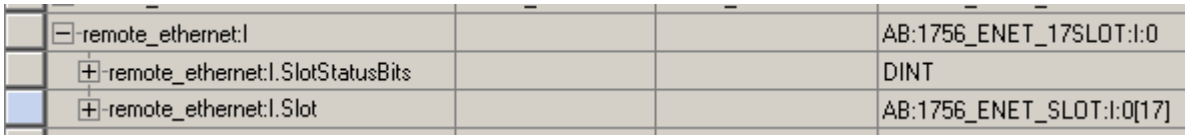
Configurazione dell'I/O distribuito Logix

Tutta la configurazione I/O viene eseguita nella struttura ad albero di progetto del software RSLogix 5000. Dalla Configurazione I/O inserire un modulo di comunicazione per il tipo di rete scelto.

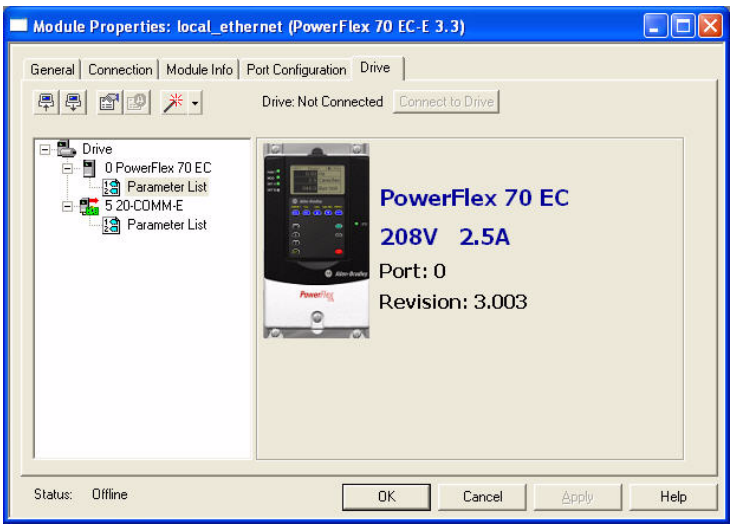
Nell'immagine è mostrata l'aggiunta di un modulo I/O 1756-IB32 remoto collegato tramite una rete EtherNet/IP.



Si noti che i tag corrispondenti al modulo I/O remoto sono stati aggiunti automaticamente al database dei tag del controllore.

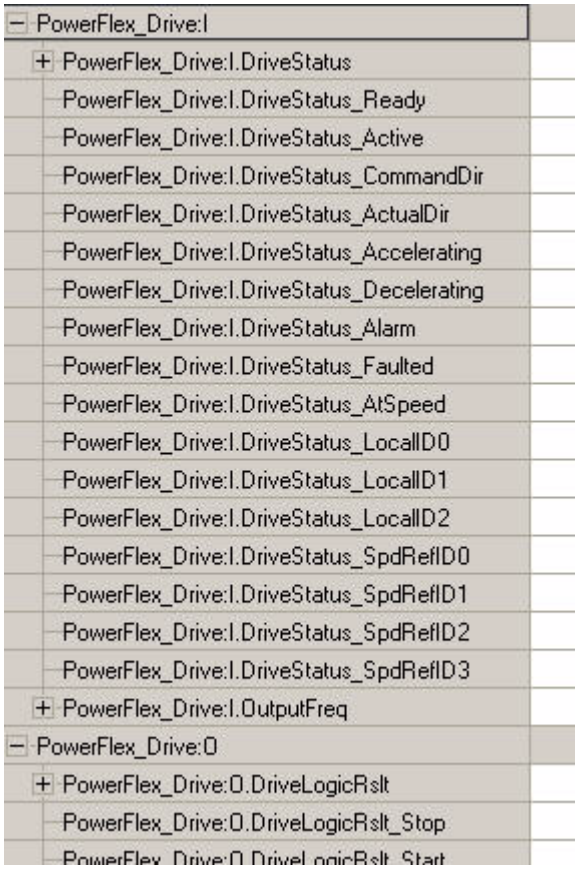


È possibile aggiungere un inverter a velocità variabile collegato in rete, come l'inverter PowerFlex, utilizzando la stessa procedura.



Anche in questo caso, il software RSLogix 5000 genererà automaticamente i nuovi tag per qualsiasi dispositivo con un profilo nel software RSLogix 5000 e collegato su una rete EtherNet/IP o ControlNet. Per la rete DeviceNet, Safety I/O GuardLogix è integrato allo stesso modo. È necessario configurare altri dispositivi DeviceNet utilizzando il software di configurazione RSNetWorx e i file EDS che funzionano fondamentalmente come il software STEP 7 Profibus Manager e i file GSD.

Di seguito sono mostrati i tag dei profili dei dispositivi nel software RSLogix 5000, disponibili per centinaia di dispositivi Rockwell Automation.



Reti

Per informazioni sulle reti, fare riferimento alle seguenti sezioni.

Reti in S7

Rete Profibus DP, DPV1, DPV3

Nel mondo S7, il tipo di rete principale per la comunicazione con i dispositivi è Profibus DP in svariate implementazioni. Alcuni controllori S7-300 di alta gamma e tutti i controllori S7-400 hanno porte master Profibus incorporate.

Rete Profibus - Altro

Profibus FMS e FDL servono per la comunicazione di dati tra i controllori. Eseguono una funzione simile alla rete Industrial Ethernet, e la configurazione è quasi identica. Le differenze consistono nella necessità di processori di comunicazione Profibus anziché della rete Ethernet e nell'utilizzo del cablaggio Profibus.

È possibile utilizzare Profibus DPv2 per eseguire il collegamento ai servoazionamenti nei controllori S7-315T e S7-317T per il controllo assi di fascia bassa.

Rete Industrial Ethernet

La rete Industrial Ethernet di Siemens è la variante Siemens della rete Ethernet in un ambiente industriale. Viene utilizzata principalmente per la comunicazione tra i controllori e per la comunicazione tra controllore e computer di programmazione.

A parte alcuni dei controllori più recenti dotati di Profinet, i controllori S7 non dispongono di porte Ethernet incorporate. Sui sistemi S7 che utilizzano Industrial Ethernet i processori di comunicazione sono montati nei rack.

In base al processore di comunicazione è possibile utilizzare i seguenti protocolli:

- S7 (protocollo proprietario per la comunicazione tra i controllori S7)
- Socket di base TCP (Transmission Control Protocol)
- ISO-on-TCP (TCP esteso con verifica aggiuntiva)
- Socket di base UDP (User Datagram Protocol)

Per gestire la maggior parte degli aspetti della comunicazione su queste reti, è necessario programmare con del codice applicativo.

Nell'ambiente Rockwell Automation, questa funzionalità può essere implementata utilizzando le porte EtherNet/IP integrate, i moduli EtherNet/IP Bridge e/o i moduli EWEB.

Profinet

Profinet fornisce una funzionalità simile a Profibus DP su una rete Industrial Ethernet con gli stessi requisiti di programmazione. Una rete che utilizza Profinet è simile a Profibus, ad eccezione di cavi e connettori diversi e all'utilizzo di moduli di interfaccia di campo Ethernet anziché Profibus. Per il collegamento alla rete sono utilizzati controllori con un'interfaccia Profinet incorporata o un processore di comunicazione adatto a Profinet.

In alternativa, è possibile collegare una rete Profibus DP esistente a Profinet, tramite un proxy o utilizzando la porta Profibus DP di un controllore che supporta Profinet.

Alcuni moduli di interfaccia di campo Profinet dispongono di più porte RJ45 con uno switch integrato per consentire una topologia di bus di tipo Profibus, se necessario.

Profinet fornisce le seguenti tre possibilità di comunicazione:

- Profinet CBA (Component Based Automation), utilizzato principalmente per la comunicazione tra controllori; utilizza l'hardware Ethernet standard e lo stack software TCP/IP.
- Profinet IO per trasferimenti schedulati, come inverter o moduli I/O; utilizza l'hardware Ethernet standard, ma bypassa lo stack software TCP/IP.
- Profinet IRT (Isochronous Real Time) per le applicazioni di controllo assi; utilizza hardware Profinet specifico, bypassa inoltre lo stack software TCP/IP e deve trovarsi su un segmento di rete protetto.

Se viene utilizzato un framework Profinet CBA, le reti Profibus, Profinet e Industrial Ethernet possono essere integrate tramite configurazione grafica, con necessità ridotta di programmazione aggiuntiva. Le reti EtherNet/IP di Rockwell Automation forniscono questa funzionalità tramite hardware standard e lo stack software TCP/IP standard con funzioni incorporate, come l'istruzione Message (MSG) e i tag prodotti/consumati.

Reti in Logix

NetLinx è il termine che identifica la soluzione Rockwell Automation nell'area delle tecnologie di rete. Di seguito sono riportate le reti principali utilizzate nei sistemi Logix:

- EtherNet/IP
- ControlNet
- DeviceNet

Queste reti presentano varie funzioni degne di nota. Sono tutte progettate in base al protocollo CIP (Common Industrial Protocol), che consente di controllare, configurare e raccogliere dati su qualsiasi rete NetLinx. Di conseguenza, i dati possono passare da una rete all'altra senza la necessità di utilizzare software di traduzione dei protocolli o proxy.

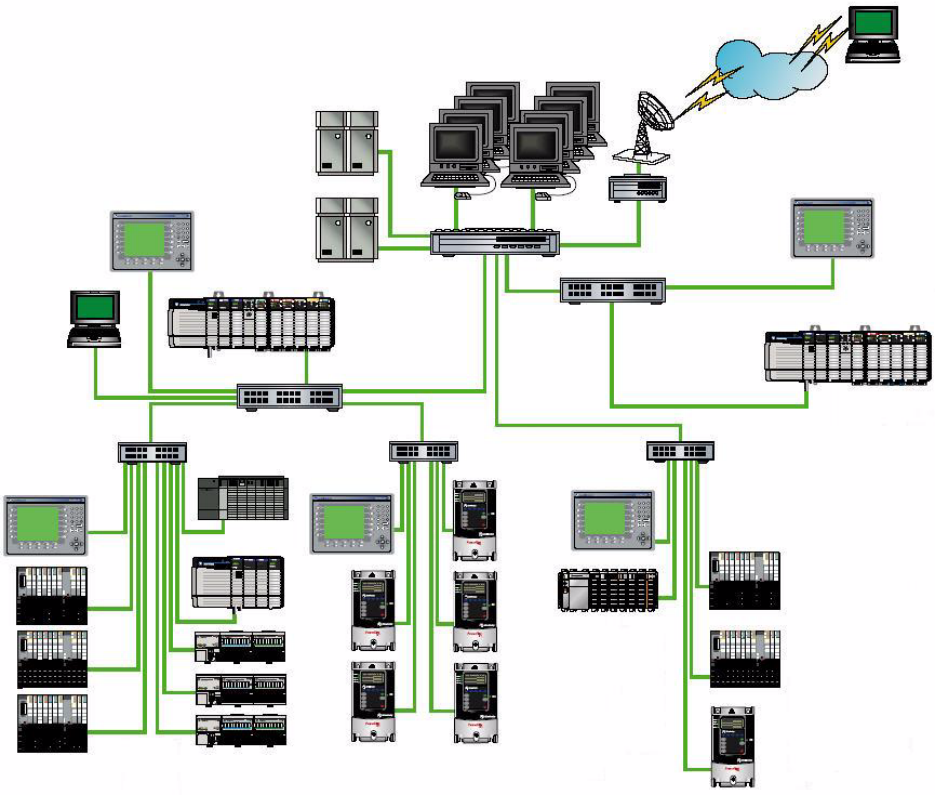
I tecnici che stanno acquisendo esperienza con i sistemi Logix potrebbero rimanere piacevolmente sorpresi dalla natura integrata e dall'accuratezza della configurazione delle reti Logix.

Rete EtherNet/IP

La rete EtherNet/IP offre una suite completa di servizi di controllo, configurazione e raccolta dati. Utilizza il protocollo TCP/IP per la messaggistica generale e/o lo scambio di informazioni e UDP/IP per la messaggistica I/O. Viene utilizzata più frequentemente nei seguenti tipi di configurazioni:

- controllo I/O generale
- scambio di dati tra controllori
- collegamento di molti computer
- collegamento di molti dispositivi
- connettività ai sistemi aziendali
- integrazione dei dispositivi di protezione
- controllo assi (in futuro).

Esempio tipico di Ethernet/IP

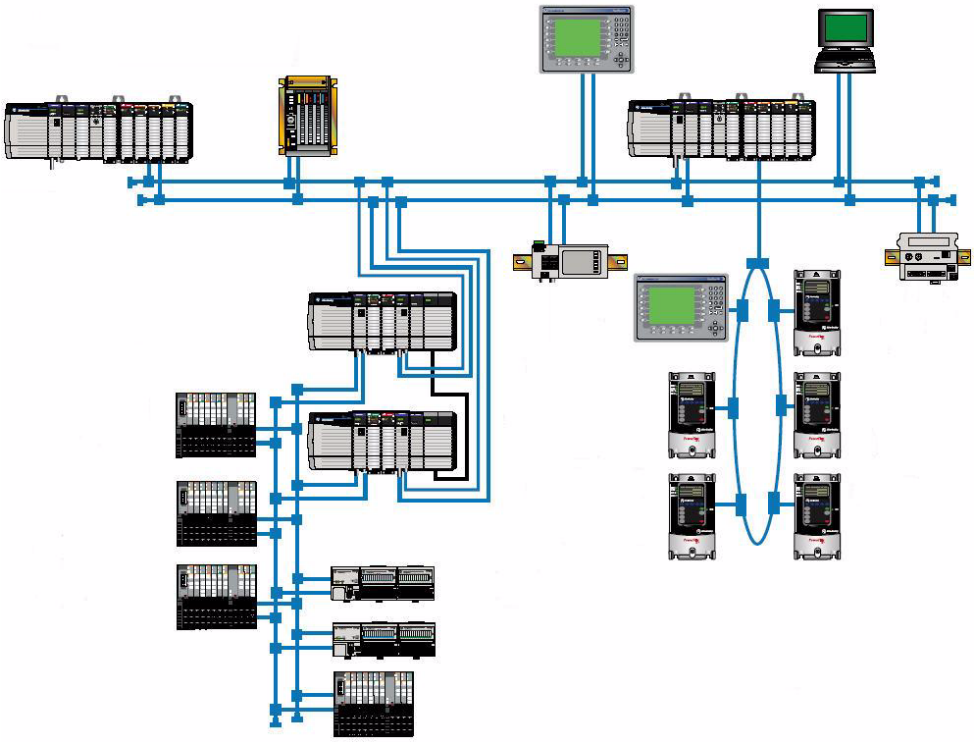


Rete ControlNet

ControlNet è una rete di controllo in tempo reale che garantisce il trasporto di dati I/O con tempo critico, dati di interblocco e messaggi, compresi upload/download dei dati di programmazione e configurazione su un unico collegamento fisico. Viene utilizzata più frequentemente nei seguenti tipi di configurazioni:

- controllo I/O generale
- scambio di dati tra controllori
- dorsale per più reti DeviceNet distribuite

Esempio tipico di ControlNet

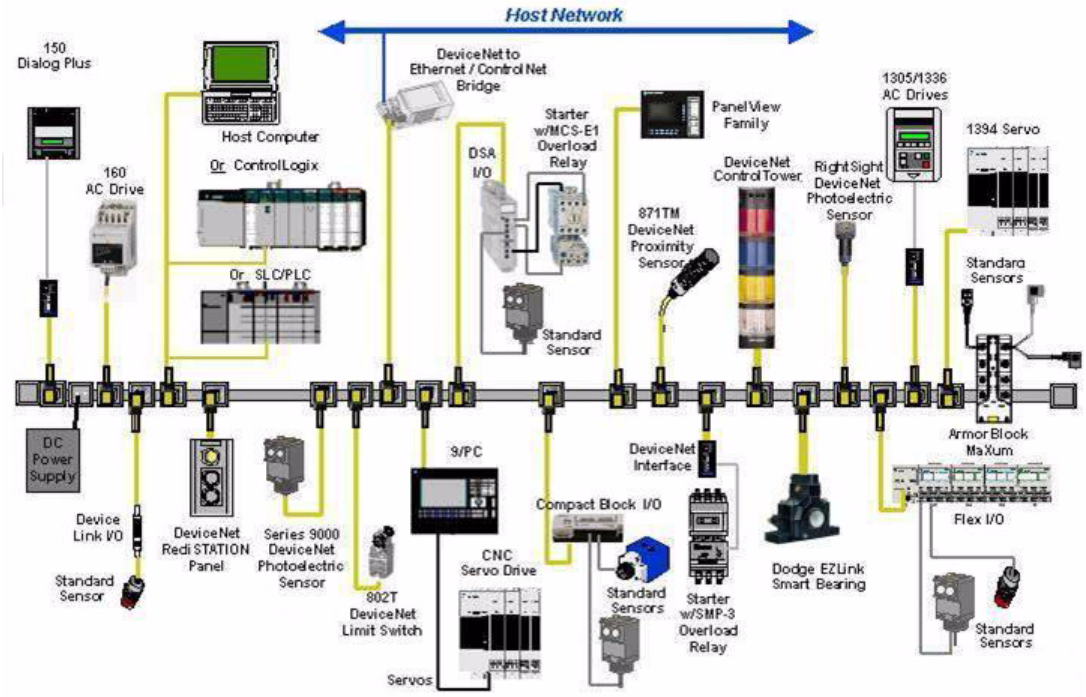


Rete DeviceNet

DeviceNet è una rete per dispositivi industriali di fascia bassa. Progettata per dispositivi con volume di dati ridotto per il funzionamento in tempo reale, viene utilizzata più frequentemente nei seguenti tipi di configurazioni:

- applicazioni contenenti dispositivi distribuiti con pochi punti
- rete di drive di terze parti e altri dispositivi di terze parti "semplici"
- sistemi nei quali i dispositivi devono essere collegati direttamente alla rete con dati e alimentazione nella stessa connessione
- quando sono necessarie informazioni di diagnostica avanzata.

Esempio tipico di DeviceNet



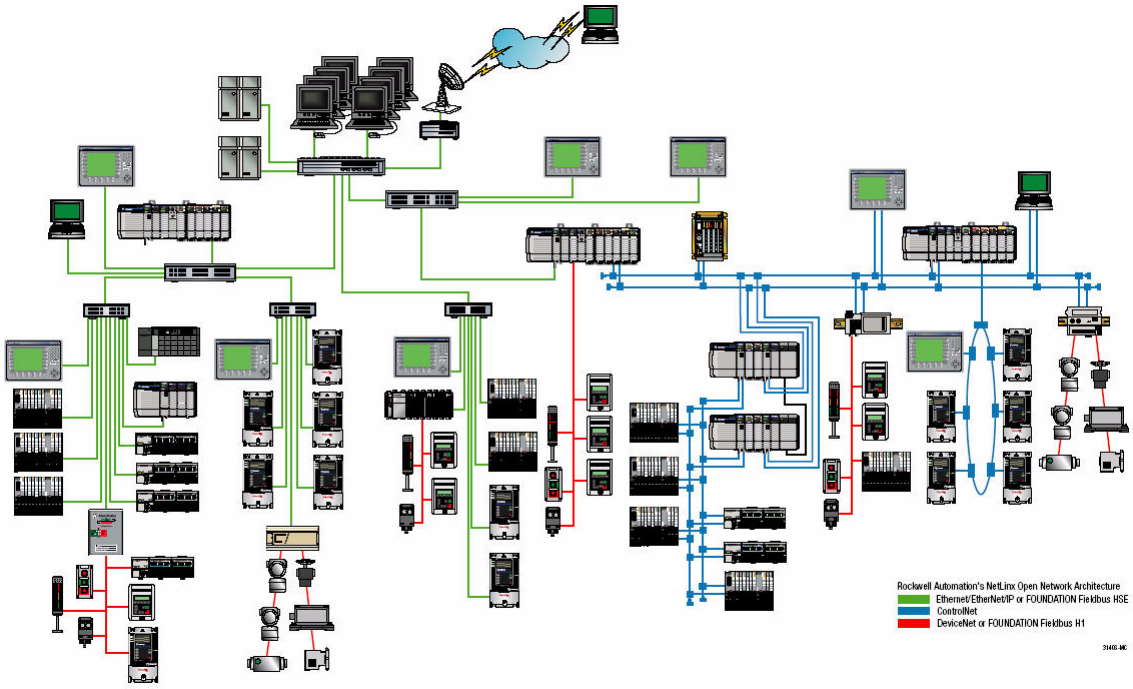
Inteconnessione delle reti NetLinx

È possibile interconnettere le reti NetLinx in due modi diversi.

- Backplane di comunicazione, che consente il collegamento di più reti contemporaneamente.
- Dispositivi di collegamento di comunicazione, che collegano due reti in modo continuo.

Con questi due metodi non sono necessari controllori né programmazione.

Esempio di un sistema di controllo basato sulle reti NetLinx


Rockwell Automation's NetLinx Open Network Architecture
Ethernet/EtherNet/IP or FOUNDATION Fieldbus HSE
ControlNet
DeviceNet or FOUNDATION Fieldbus H1

Conversione dell'interfaccia operatore

Fare riferimento all'Appendice B.

Conversione dei sistemi contenenti controllori distribuiti

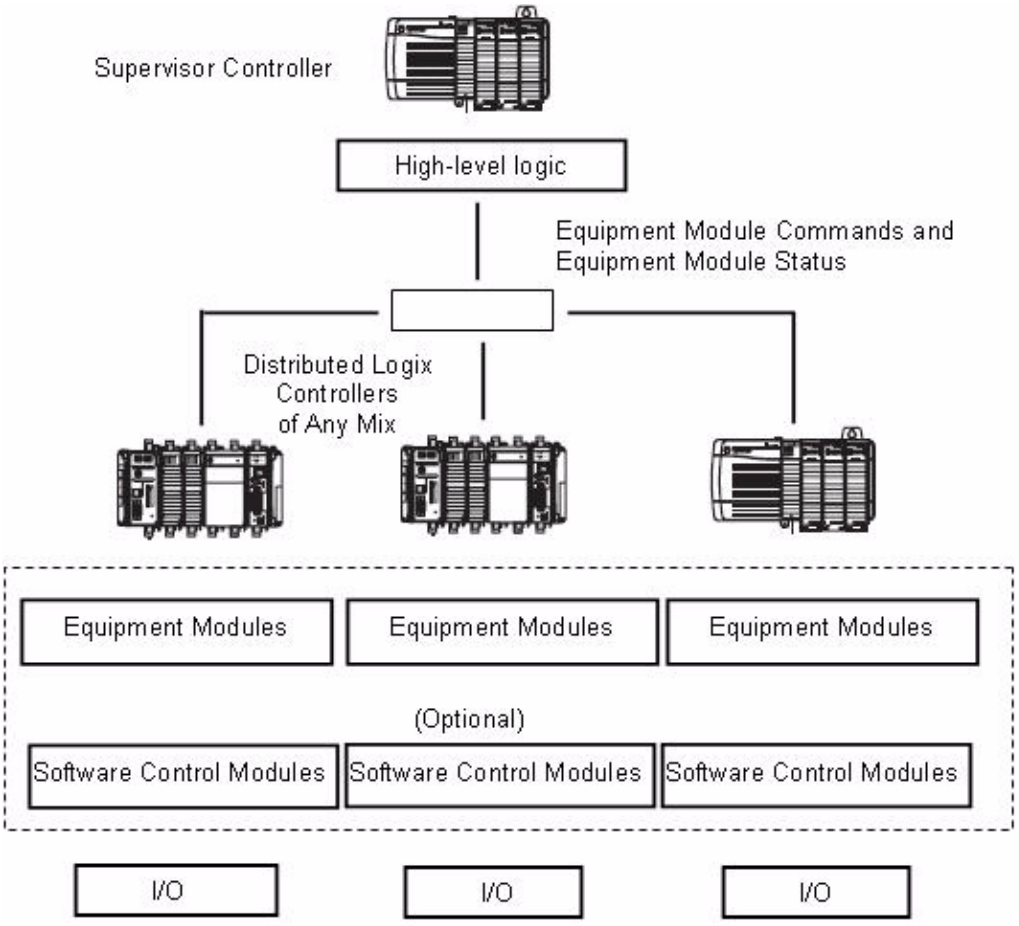
In questa sezione viene illustrato quanto segue:

- in che modo si costruisce un'applicazione di controllo discreto contenente un gruppo di unità funzionali utilizzando più controllori
- in che modo un metodo simile può essere utilizzato in un'applicazione per il controllo di processo progettata in base allo standard S88.

Implementazione di hardware e software

Controllo discreto generale

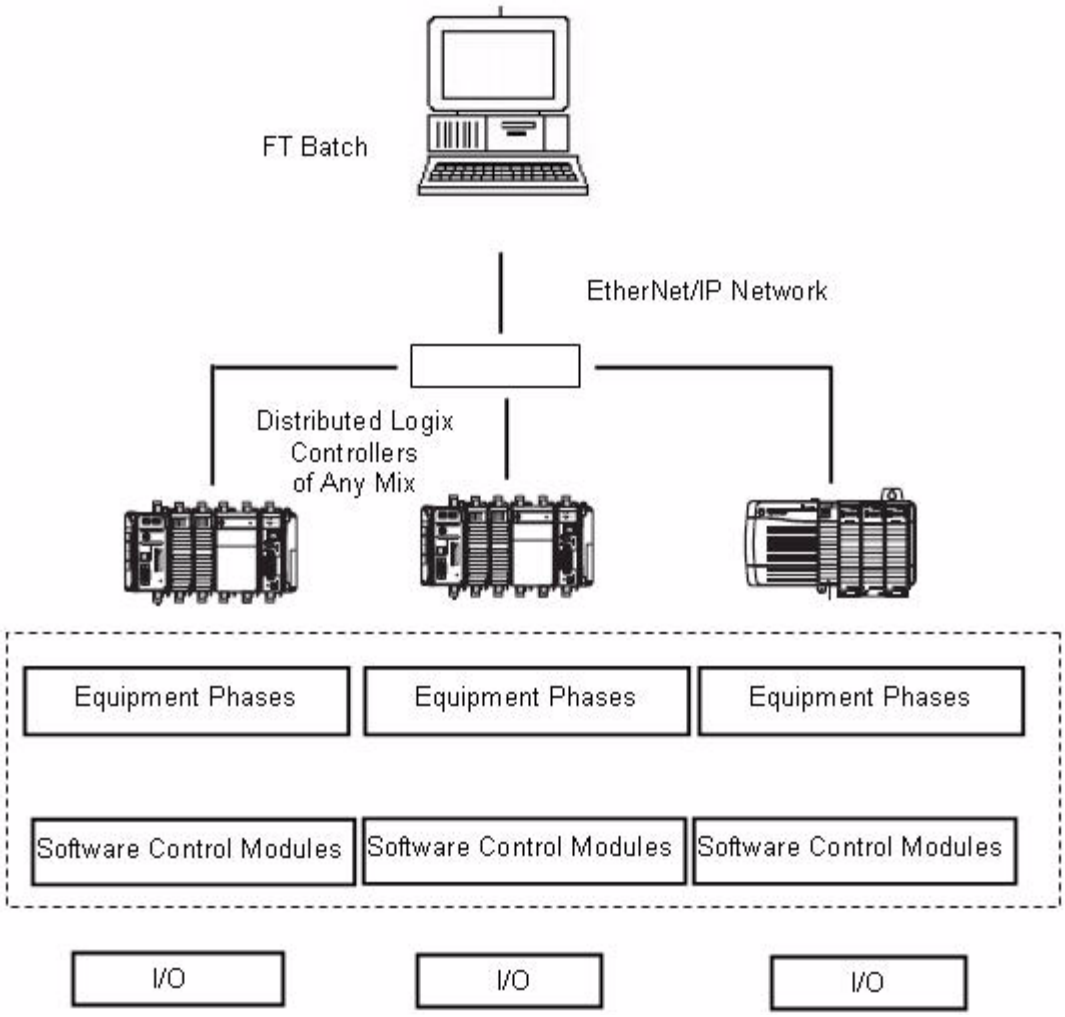
Di seguito è mostrato il modello hardware e software per la logica distribuita per il controllo discreto generale. In questo caso, il ruolo di supervisore sarà ricoperto da un controllore. Per collegare i controllori, è possibile utilizzare la rete EtherNet/IP o ControlNet. Il modello produttore/consumatore e i messaggi espliciti possono essere utilizzati per scambiare i dati all'interno del sistema.



Controllo di processo

Nello schema di seguito è mostrata la struttura hardware e software per un'applicazione per il controllo di processo S88. Sul PC sarà eseguito FactoryTalk Batch, che è un pacchetto software per eseguire batch di produzione per mezzo di ricette. Il software FactoryTalk Batch si trova in un PC e comunica con ciascun controllore tramite la rete EtherNet/IP.

Le fasi delle apparecchiature sono configurate in PhaseManager come descritto successivamente nel Capitolo 2. Eseguono la logica di fase e comunicano con l'I/O del sistema di controllo tramite moduli di controllo.



Collegamento di dispositivi Siemens e Rockwell Automation

In alcune circostanze può essere necessario collegare fra loro le apparecchiature Siemens e Rockwell Automation. Si consiglia di utilizzare prodotti di società partner facenti parte del programma Encompass.

Controllori

I controllori Logix possono essere connessi alle reti S7 utilizzando:

- moduli in-rack
- gateway di comunicazione autonomi.

Dispositivi distribuiti

Alcune sistemi I/O, inverter PowerFlex e terminali HMI di Rockwell Automation sono collegati a Profibus tramite schede di comunicazione, interfacce incorporate o moduli di interfaccia.

Funzioni di Logix che potrebbero non essere familiari agli utenti di S7

Introduzione

In questo capitolo sono descritte le funzioni di Logix che potrebbero non essere familiari agli utenti S7.

Argomento	Pagina
Confronto tra blocchi organizzativi (OB) di S7 e task Logix	36
Tag senza indirizzi	47
I/O e tag alias	51
Linguaggi di programmazione	53
Istruzioni add-on	57
Common Industrial Protocol (CIP)	58
Scambio di dati tra controllori	60
Tipi di dati definiti dall'utente	61
Aggiornamento asincrono degli I/O	62
Tipo di dati DINT	62
Phase Manager	63
Tempo di sistema coordinato (CST)	65
Ingressi con registrazione cronologica	65
Uscite schedulate	65
Nessuna variabile temporanea	66
Non sono necessari accumulatori o registri speciali	66

Alcune funzioni del sistema Logix sono più facili da utilizzare e gestire rispetto a quelle di S7. Ad esempio, i dati sono organizzati in database di tag senza indirizzi assoluti, mentre in S7 i dati hanno indirizzi assoluti selezionati dal programmatore in aree di memoria definite.

Per altri aspetti, la struttura di Logix è abbastanza simile a quella di S7, ma è presentata in modo diverso, ad esempio la struttura interna dei task è simile ai blocchi organizzativi di S7.

In questo capitolo vengono analizzate le funzioni diverse (come i tag) e vengono confrontate le funzioni essenzialmente simili (come i task).

L'obiettivo è:

- fornire all'utente di S7 che esegue la conversione in Logix informazioni che renderanno più facile e veloce il processo di progettazione
- illustrare le funzionalità di Logix, in modo che i tecnici non tentino di ricreare ciò che esiste già all'interno del firmware del controllore.

Confronto tra blocchi organizzativi (OB) di S7 e task Logix

Questo confronto tra i blocchi organizzativi e i task illustra la struttura di un programma Logix all'utente di S7.

I blocchi organizzativi e i task sono simili poiché entrambi sono richiamati dal sistema operativo del controllore anziché dal programma utente. In STEP 7 (e Logix), esistono tre tipi di blocchi organizzativi (task in Logix).

- OB ciclo del programma (task continuo in Logix) in cui l'OB ricomincia dall'inizio al termine del ciclo.
- OB interrupt ciclico (task periodico in Logix) in cui l'OB viene eseguito a intervalli di tempo preconfigurati.
- OB interrupt hardware (task evento in Logix) eseguiti in risposta a uno stimolo hardware.

Molti programmatori di STEP 7 non utilizzano gli OB interrupt ciclico.

Logix fornisce un sistema operativo multi-tasking configurabile dall'utente che consente alla CPU di allocare risorse in base alle necessità dell'applicazione

Blocchi organizzativi in S7

Il tipo di OB è definito dal relativo numero. I blocchi organizzativi sono eseguiti continuamente (solo OB1), periodicamente (OB30 – OB38), possono essere eseguiti in base agli eventi (OB40 – OB47) oppure possono essere eseguiti quando si verificano determinati errori. Con Logix, i task non sono numerati, ma vengono identificati da un nome definito dall'utente.

Se necessario, è possibile assegnare un nome significativo a un OB di STEP 7.

OB1 ciclo del programma

OB1 viene eseguito in un ciclo continuo. Al termine dell'esecuzione, i valori della tabella immagine delle uscite sono inviati alle uscite, la tabella immagine degli ingressi viene aggiornata in base alle uscite e OB1 viene riavviato.

Un programma STEP 7 non deve necessariamente includere OB1, tuttavia se è incluso, sarà eseguito continuamente.

Frammento OB1 tipico:

```
Network 3 : Title:
callup valve and motor control module

          CALL  "ValveMotor_Calls"

Network 4 : Title:
callup switch control module

          CALL  "Switch_Calls"

Network 5 : Title:
callup flow totalisers control module

          CALL  "Totaliser_calls"

Network 6 : Title:
callup analogue input control module

          CALL  "AnalogueIn_calls"
```

OB1 è la radice della gerarchia di chiamate per tutto il codice eseguito continuamente.

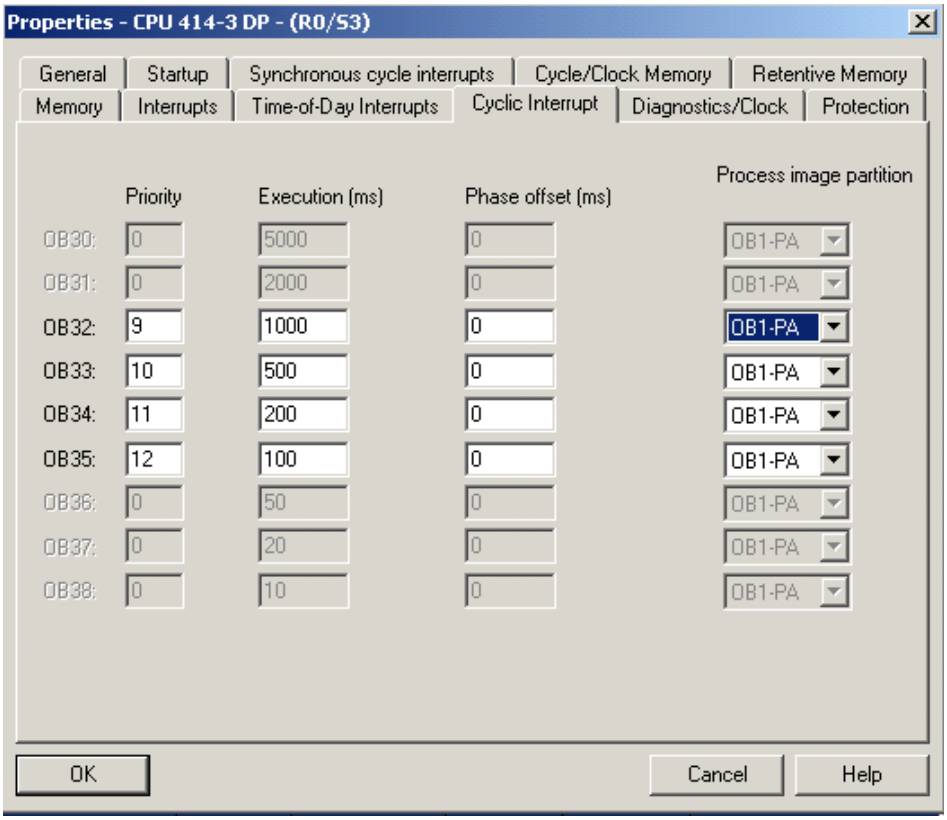
OB1 è simile al **task continuo** di Logix (naturalmente ne può essere presente solo uno). Nella terminologia S7, OB1 è descritto come “ciclo di programma”.

Per i lettori più esperti di Logix che di STEP 7, può essere utile sapere che nella logica ladder di STEP 7 una rete corrisponde a un ramo Logix. Nell'elenco di istruzioni di STEP 7, le reti sono comunque presenti ma servono solo a migliorare la struttura del codice. Suddividono il codice in sezioni e consentono l'aggiunta di commenti. Tutto il codice può essere inserito in una rete, se necessario, e viene compilato ed eseguito perfettamente.

OB30 – OB38 interrupt ciclici

Questi OB vengono eseguiti a intervalli fissi configurabili. È inoltre possibile configurarne la priorità. Gli OB a priorità più alta interrompono eventuali OB in esecuzione con priorità inferiore.

Modalità di configurazione degli OB richiamati periodicamente



Il numero di OB periodici disponibili varia in base al tipo di controllore. Il numero a priorità più bassa rappresenta la priorità di interrupt più alta (la selezione della priorità è disponibile solo con i controllori S7 400). L'esecuzione (ms) è il periodo di esecuzione dell'OB. L'offset di fase consente di sfasare l'attivazione degli interrupt periodici uno rispetto all'altro. La selezione della partizione dell'immagine di processo consente di suddividere la tabella immagine I/O e di aggiornare tale partizione solo quando si verifica l'interrupt (questa funzione è disponibile solo nei controllori S7 400). L'impostazione predefinita è l'intera tabella. In Logix, vedere la selezione di aggiornamento degli I/O del task e i comandi IOT.

In genere, il contenuto di un OB interrupt periodico è simile al contenuto di OB1. È costituito da chiamate di funzioni e blocchi funzione eseguiti in base alla periodicità dell'OB.

Questi OB sono simili ai **task periodici** in Logix. Nella terminologia S7, OB30 – OB38 sono chiamati OB interrupt ciclici.

OB40 – OB47 OB interrupt hardware

Questi OB possono essere configurati per attivarsi in casodi un evento di ingresso. È inoltre possibile configurarne la priorità.

Corrispondono ai **task evento** in Logix. Nella terminologia S7, OB40 – OB47 sono chiamati interrupt hardware.

Ad esempio, l'evento hardware più semplice che potrebbe essere gestito da un OB interrupt hardware (o task evento) è una variazione di stato di un ingresso digitale. Un interrupt hardware (o task evento) garantisce una risposta molto rapida al cambiamento.

I task evento sono più flessibili degli OB interrupt hardware, con trigger non solo dall'I/O, ma anche dagli eventi di rete, dalle istruzioni di programmazione e dagli eventi di movimento.

Struttura di programmazione in STEP 7

Un programma tipico include blocchi organizzativi (OB), blocchi funzione (FB), funzioni (FC) e blocchi dati (DB). In genere, sono presenti blocchi funzione del sistema (SFB) e funzioni di sistema (SFC).

- Dai blocchi organizzativi (ciclo di programma, interrupt ciclico o entrambi) vengono eseguite le chiamate dei blocchi funzione e delle funzioni.
- Un blocco funzione contiene codice ed è associato a un blocco dati che contiene i dati statici richiesti dal blocco funzione (FB). Oltre ai dati statici, l'FB include dati temporanei. Gli FB sono utilizzati quando la logica deve conservare i valori tra un'esecuzione e l'altra.
- Una funzione contiene codice ma non dati statici. Include dati temporanei. Le FC sono utilizzate quando la logica viene completata in un'unica esecuzione, e non è necessario conservare i valori.
- I blocchi dati sono aree per la memorizzazione dei dati statici. Saranno descritti nella sezione successiva.
- SFB e SFC sono blocchi funzione del sistema e funzioni di sistema. Possono essere copiati dalle librerie incluse in un'installazione STEP 7 e inseriti in un progetto.
- Dopo avere eseguito questa operazione, possono essere chiamati da qualsiasi parte del programma.

In STEP 7 non esiste una struttura equivalente a Programma/Routine di Logix. L'OB è la radice della catena di chiamate di FB e FC, tuttavia la modalità di esecuzione di questa operazione viene stabilita dal programmatore.

Task in Logix

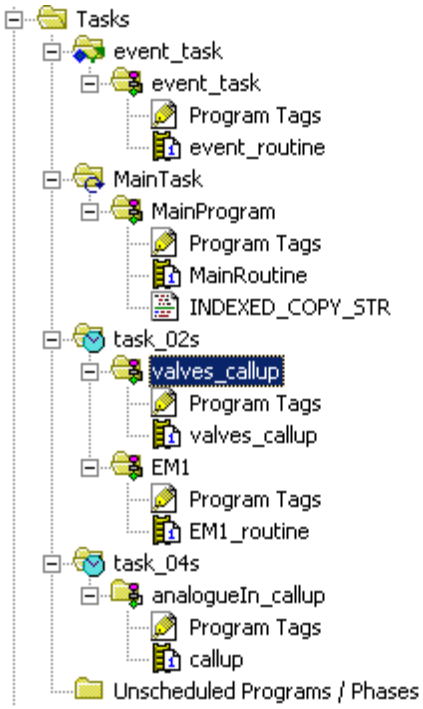
I task sono richiamati dal sistema operativo. Un task fornisce schedulazione e priorità per uno o più programmi. Ciascun programma contiene una sezione di dati e una o più routine di codice.

I task possono essere periodici, ad evento o continui. A ciascun task può essere assegnata una priorità. Il task continuo, se presente, ha sempre la priorità più bassa.

Un progetto Logix includerà un task con il nome predefinito di MainTask. Questo task può essere continuo, periodico o ad evento. È possibile modificarne il nome, se necessario.

Struttura di task e programmi in Logix

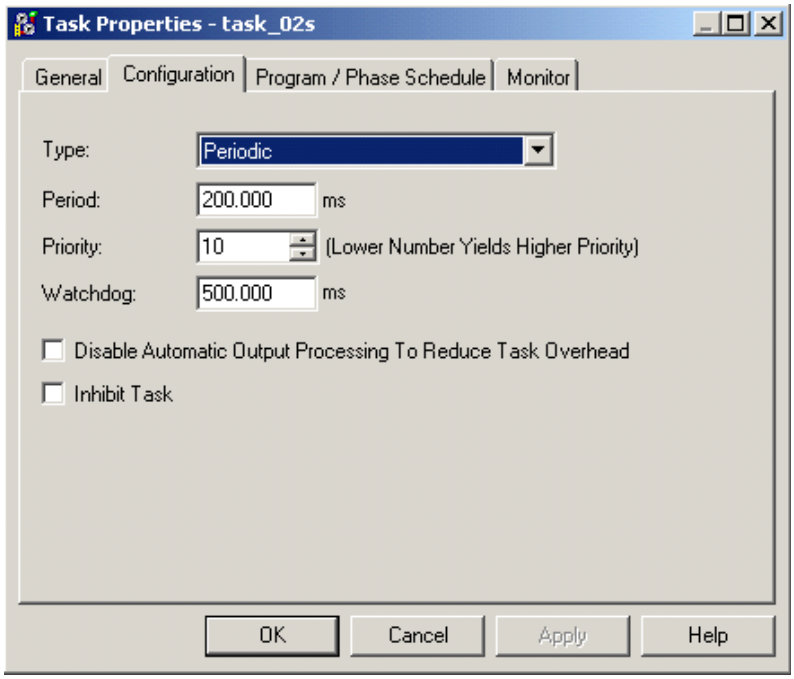
L'immagine di una struttura ~~da~~ albero esemplificativa di RSLogix 5000 consente di illustrare in che modo task e programmi sono strutturati.



Nell'immagine riportata sopra, l'icona a sinistra di “event_task” indica un task evento. L'icona a sinistra di “MainTask” indica un task continuo, mentre l'icona a sinistra di “task_02s” indica un task periodico.

Task periodici

I task periodici si attivano a un intervallo configurato costante. Di seguito è mostrata la configurazione del periodo e della priorità.



La configurazione è simile alla pagina di configurazione di OB30 – OB38 descritta nella sezione "OB30 – OB38 interrupt ciclici".

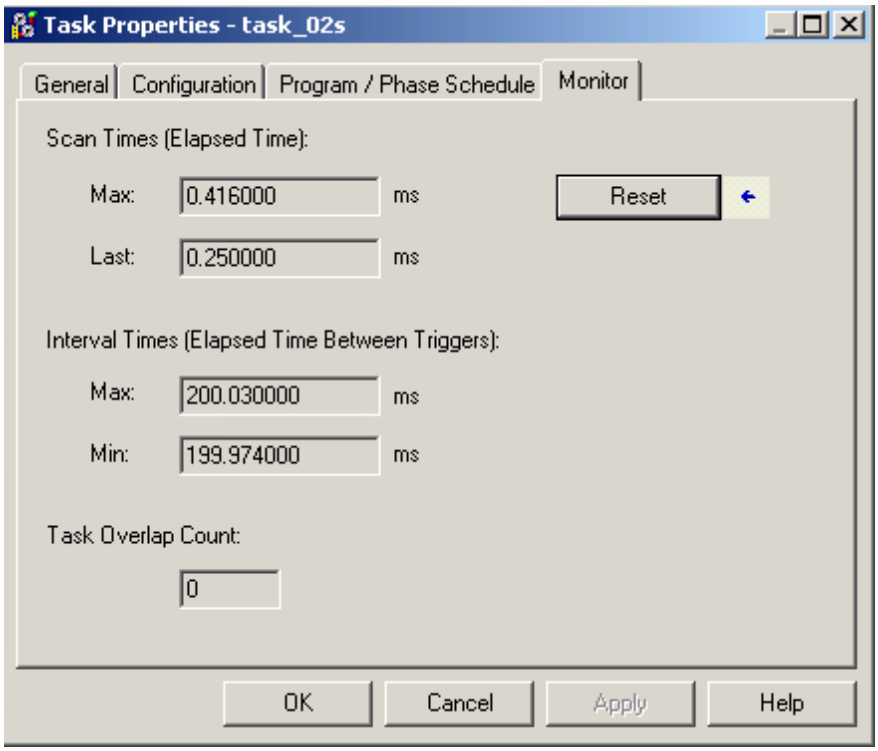
Schedulazione di task periodici

Lo scopo del sistema dei task è:

- consentire al programmatore di scegliere le frequenze appropriate per l'esecuzione dei programmi. Se il codice non viene eseguito più frequentemente del necessario, la potenza della CPU del controllore viene utilizzata più efficacemente per le priorità dell'applicazione.

- utilizzare il sistema di priorità per consentire ai task critici di interrompere quelli con priorità più bassa, facilitandone l'esecuzione alla frequenza prevista.

Questi tempi possono essere controllati facilmente da Task Properties/Monitor.



Cosa accade se si verifica un trigger quando un task è in esecuzione?

- Se il nuovo trigger è relativo a un task con priorità più alta rispetto a quello in esecuzione, il task in esecuzione viene interrotto da quello nuovo e verrà riavviato al termine del task con priorità più alta.
- Se il nuovo trigger è relativo a un task con priorità più bassa rispetto a quello in esecuzione, il task in esecuzione continuerà e il nuovo task attenderà finché non sono più in esecuzione task con priorità più alta.
- Se il nuovo trigger è relativo a un task con la stessa priorità di quello in esecuzione, il controllore eseguirà entrambi i task passando da uno all'altro a intervalli di 1 ms.
- Se il nuovo trigger è relativo allo stesso task di quello in esecuzione, il nuovo trigger sarà ignorato. Si tratta di una condizione di **sovrapposizione**.

Il numero di sovrapposizioni che si sono verificate dall'ultimo azzeramento del contatore è mostrato nella finestra delle proprietà del task. Un numero diverso da zero indica che è necessario aumentare il periodo di interrupt.

CONSIGLIO Evitare di commutare i task inutilmente per evitare di sprecare potenza di elaborazione per le commutazioni non necessarie.

Quando si programmano interrupt periodici in Logix, si notino le seguenti similitudini e differenze con STEP 7:

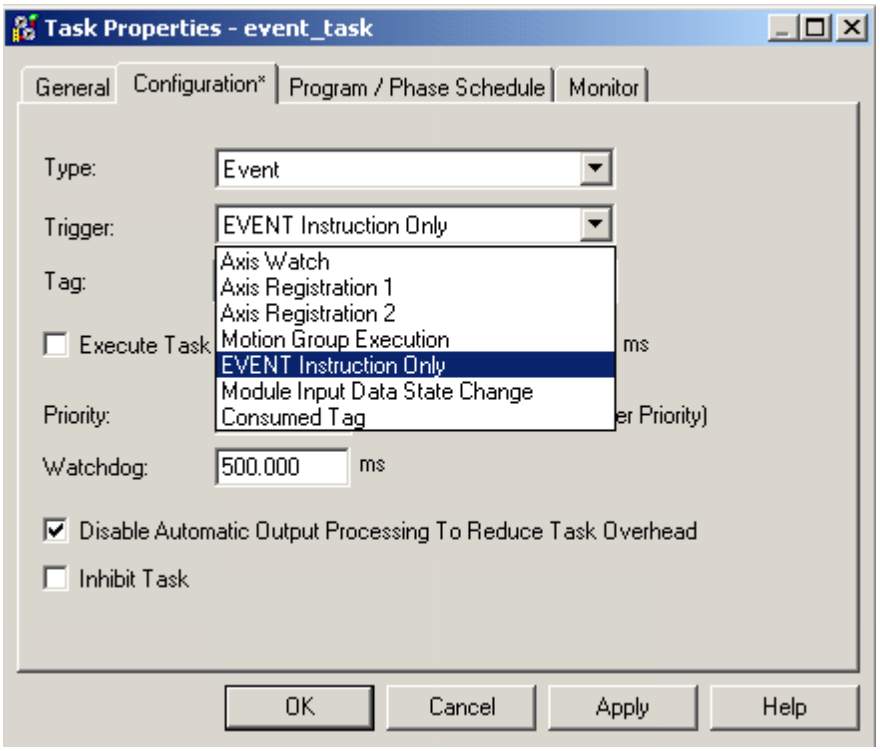
- In STEP 7, le chiamate saranno eseguite dall'OB configurato per eseguire alla frequenza scelta le funzioni e i blocchi funzione desiderati a tale frequenza. In Logix, i programmi e le routine sono inserite nella struttura ad albero del progetto del task.
- In STEP 7 e in Logix, il codice applicativo effettivo non sarà molto diverso dal codice in un task continuo. Si noti che la frequenza costante e nota di un task periodico consente ai programmatori di trasformare un semplice incremento di variabile in un timer.
- In entrambi i sistemi sarà necessario verificare la presenza di sovrapposizioni mentre si sviluppa e si esegue il test del codice. Il tempo di esecuzione dell'OB o del task deve essere molto inferiore rispetto al relativo periodo di esecuzione.
- Controllare il tempo di esecuzione dei task Logix è semplice. Utilizzare la schermata delle proprietà del task mostrata sopra. In STEP 7 sarà necessario campionare l'orologio del sistema all'inizio e alla fine dell'OB, sottrarre i valori e memorizzare il risultato in una variabile per il monitoraggio.

- In un controllore S7, le sovrapposizioni causano l'interruzione del controllore, a meno che non venga aggiunto un OB errori per filtrare l'errore. Logix è meno rigido e conteggia solo il numero di sovrapposizioni.

- In STEP 7, è possibile sfasare l'esecuzione degli OB periodici uno rispetto all'altro. Questa opzione non è disponibile con i task Logix.

Task eventi

I task eventi saranno eseguiti quando si verifica un evento trigger configurato. In genere, hanno una priorità più alta rispetto ai task periodici.



Un task evento viene configurato aprendo la pagina delle proprietà del task e selezionando il tipo di evento. È possibile utilizzare svariati tipi di trigger dei task evento per i diversi controllori Logix.

Task continui

Un controllore Logix supporta un task continuo, tuttavia un progetto non deve includere necessariamente il task continuo. Se necessario, è possibile eseguire l'intero programma nei task periodico e ad evento.

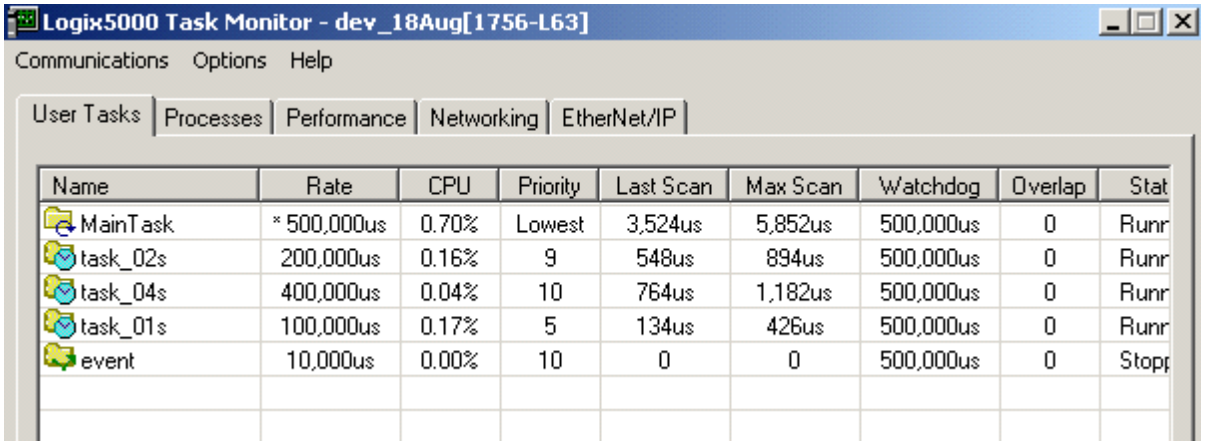
È possibile scegliere se eseguire gli aggiornamenti delle uscite da parte del task continuo al termine dell'esecuzione.

Se necessario, è possibile regolare la percentuale di tempo della CPU impiegato per comunicazioni non schedulate come percentuale di tempo dedicato al task continuo.

Monitoraggio task

Il software RSLogix 5000 include uno strumento di monitoraggio dei task (Task Monitor) che può essere un valido contributo per analizzare i task schedulati e molto altro.

L'immagine riportata di seguito illustra in che modo è possibile visualizzare i task del controllore in una tabella.



Nelle altre schede sono fornite altre informazioni dettagliate a livello di sistema sulle prestazioni del controllore. Lo strumento è incluso nel disco di installazione di RSLogix 5000.

Tag senza indirizzi

Una delle maggiori differenze che un utente di S7 noterà quando inizia a lavorare con Logix è che i dati non hanno indirizzi. I dati sono creati in un database di tag e gli indirizzi vengono assegnati automaticamente dal software RSLogix 5000. In questo modo, non è necessario che gli utenti comprendano e gestiscano gli indirizzi di memoria. In questa sezione viene descritta l'allocazione dei dati nei due sistemi.

Aree di dati in S7

Aree di dati nei controllori S7

Area indirizzi	Notazione S7	Dimensione unità
Tabella dell'immagine di processo degli ingressi	I	Bit di ingresso
	IB	Byte di ingresso
	IW	Parola di ingresso
	ID	Doppia parola di ingresso
Tabella dell'immagine di processo delle uscite	Q	Bit di uscita
	QB	Byte di uscita
	QW	Parola di uscita
	QD	Doppia parola di uscita
Memoria bit	M	Bit di memoria
	MB	Byte di memoria
	MW	Parola di memoria
	MD	Doppia parola di memoria
Timer	T	
Contatori	C	
Blocco dati	DBX	Bit di dati
	DBB	Byte di dati
	DBW	Parola di dati

Nelle seguenti sezioni vengono fornite ulteriori informazioni sulle due aree utilizzate più frequentemente nella programmazione: memoria bit e blocchi dati.

Memoria bit

Le posizioni della "memoria bit" sono indicate da M*x* dove, ad esempio:

- **M5.3** è un bit.
- **MB6** è un byte (BYTE).
- **MW8** è una parola a 16 bit (WORD).
- **MD10** è una parola a 32 bit (DWORD).

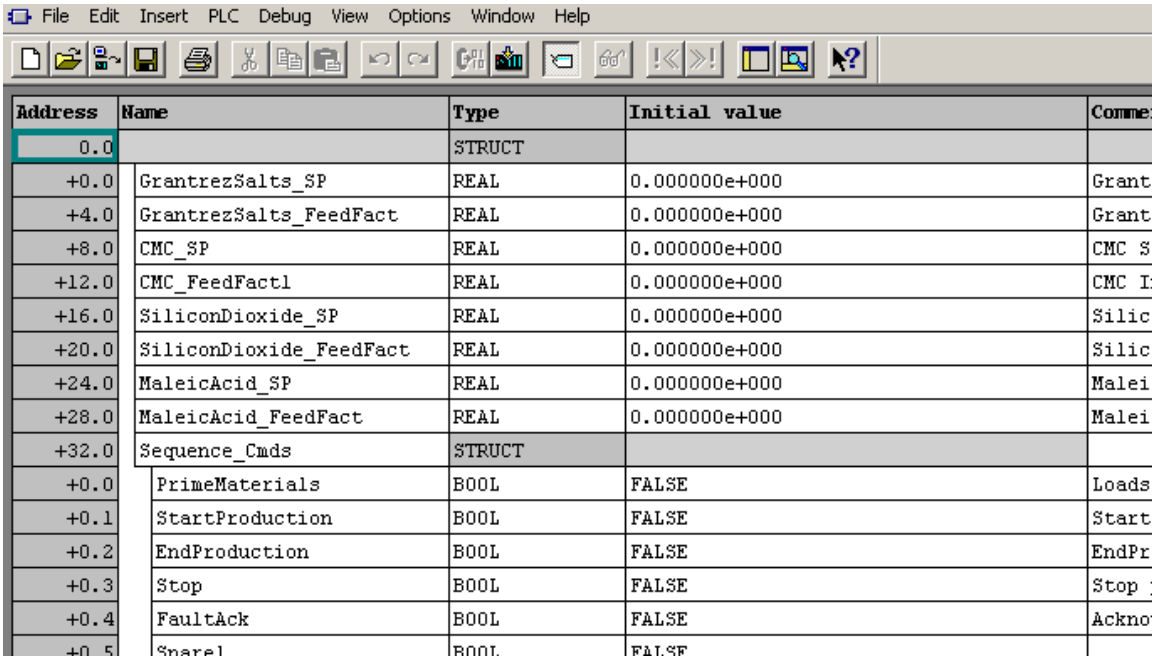
Le posizioni della memoria bit possono essere etichettate nella tabella dei simboli (simile alla tabella dei simboli PLC-5 o SLC), come mostrato nella seguente immagine.

Status	Symbol	Address	Data type	Comment
	EXT_ZONE2_ON	Q 28.2	BOOL	EXTRUDER ZONE2 ON
	EXT_ZONE3_ON	Q 28.3	BOOL	EXTRUDER ZONE3 ON
	EXT_ZONE4_ON	Q 28.4	BOOL	EXTRUDER ZONE4 ON
	EXT_ZONE5_ON	Q 28.5	BOOL	EXTRUDER ZONE5 ON
	FALSE	M 0.0	BOOL	
	FlowTotaliser	UDT 10	UDT 10	
	FSL24001	I 10.0	BOOL	
	FSL24002	I 41.0	BOOL	
	FSL24003	I 10.6	BOOL	
	FT24001	PIW 530	INT	
	FT24002	PIW 526	INT	
	FT24006	PIW 570	INT	
	GET_INDEXED_REFE...	FC 111	FC 111	
	Global_Data	DB 99	DB 99	
	IFIX_alarms	DB 71	DB 71	
	INDEXED_COMPARE	FC 102	FC 102	
	INDEXED_COPY	FC 101	FC 101	
	Interlocks_Handler	FB 70	FB 70	
	Interrupt_Execution	OB 35	OB 35	
	JUNK_BIT	M 0.2	BOOL	
	KTRON_CmdsFromM...	DB 33	DB 33	Commands & Setpoints From Manufacturing PLC
	KTRON_StatusToMan...	DB 32	DB 32	Feeder Status to ManufacturingPLC
	LF24001A	Q 17.0	BOOL	PIG LAUNCH VALVE
	LF24002A	Q 18.5	BOOL	PIG LAUNCHER
	LF24003A	Q 10.4	BOOL	PIG LAINCHER

Blocchi dati

Lo stato dei blocchi dati è simile a quello degli altri blocchi (blocchi organizzativi, blocchi funzione e funzioni), ad eccezione del fatto che contengono dati anziché codice di programmazione. La memoria nei blocchi dati è statica: i dati conservano il proprio valore finché non viene modificato.

Esempio di un blocco dati



Address	Name	Type	Initial value	Comme
0.0		STRUCT		
+0.0	GrantrezSalts_SP	REAL	0.000000e+000	Grant
+4.0	GrantrezSalts_FeedFact	REAL	0.000000e+000	Grant
+8.0	CMC_SP	REAL	0.000000e+000	CMC S
+12.0	CMC_FeedFactl	REAL	0.000000e+000	CMC I
+16.0	SiliconDioxide_SP	REAL	0.000000e+000	Silic
+20.0	SiliconDioxide_FeedFact	REAL	0.000000e+000	Silic
+24.0	MaleicAcid_SP	REAL	0.000000e+000	Malei
+28.0	MaleicAcid_FeedFact	REAL	0.000000e+000	Malei
+32.0	Sequence_Cmds	STRUCT		
+0.0	PrimeMaterials	BOOL	FALSE	Loads
+0.1	StartProduction	BOOL	FALSE	Start
+0.2	EndProduction	BOOL	FALSE	EndPr
+0.3	Stop	BOOL	FALSE	Stop
+0.4	FaultAck	BOOL	FALSE	Ackno
+0.5	Spare1	BOOL	FALSE	

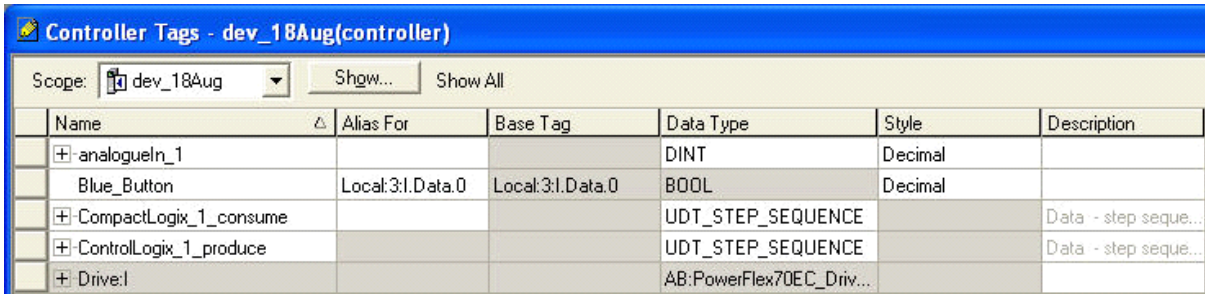
I simboli del blocco dati non sono visualizzati nella tabella dei simboli, al differenza del nome del blocco dati.

I blocchi dati possono essere assegnati per conservare i dati utilizzati dai blocchi funzione. Questi vengono chiamati blocchi dati di istanza.

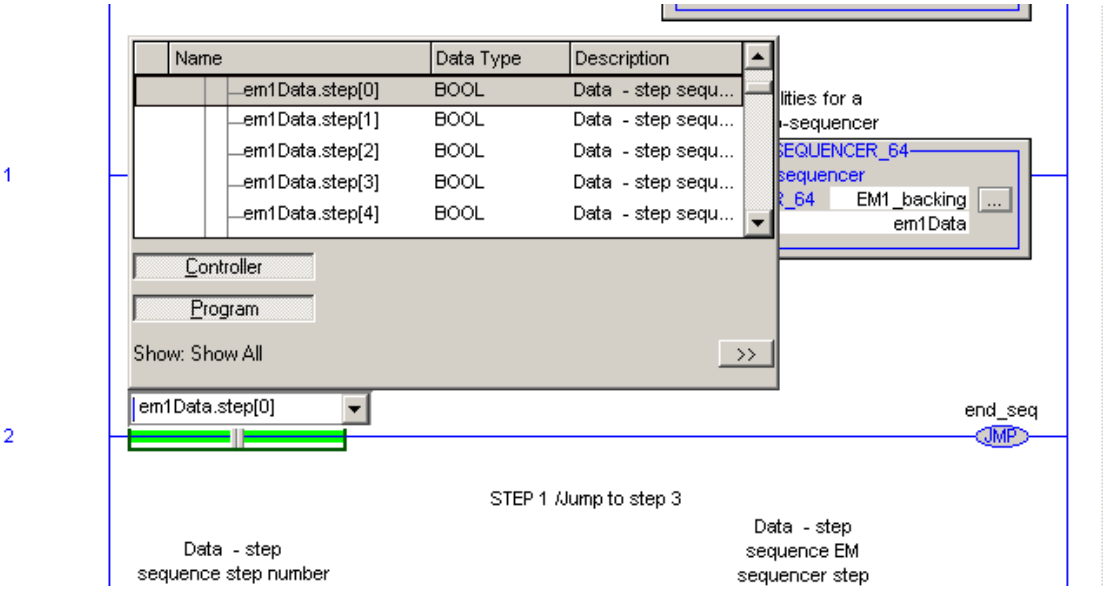
Dati in Logix

Nell'ambiente di programmazione RSLogix 5000, i dati sono configurati in un database di tag. Gli indirizzi di memoria sono nascosti, facilitando il lavoro del programmatore.

Database di tag



Selezionare un tag dal menu a discesa durante la programmazione



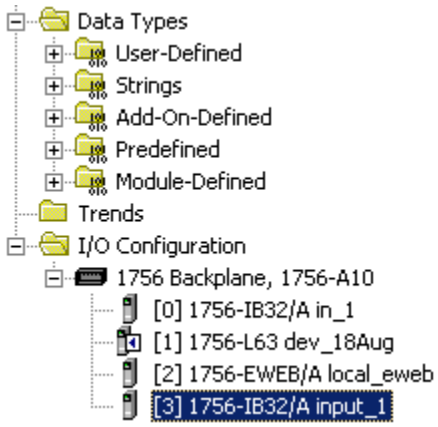
In Logix, sono presenti un **database di tag del controllore** e dei **database di tag del programma** associati a ciascun programma.

- I tag nel database del controllore sono globali ed è possibile accedervi tramite le routine in qualsiasi parte del programma.
- È possibile accedere ai tag del programma tramite le routine all'interno del programma stesso.

I/O e tag alias

Un tag alias consente di rappresentare un altro tag, mentre entrambi i tag condividono lo stesso valore. Uno degli scopi degli alias è fare riferimento ai tag I/O come descritto di seguito.

I moduli I/O possono essere aggiunti a un progetto aggiungendo il modulo al backplane del controllore nella cartella di progetto.



In questo caso, è stata aggiunta una scheda di ingresso a 32 punti ~~dallo~~ slot 3. Il numero di slot è racchiuso tra parentesi quadre all'inizio della riga. "1756-IB32/A" è il codice prodotto della scheda. "input_1" è un nome per la scheda configurato quando la scheda viene aggiunta al rack per la prima volta.

Dopo che è stata aggiunta la scheda, il software RSLogix 5000 genera automaticamente i tag di profilo del dispositivo nel database di tag nell'ambito controllore. Si tratta dei tag dell'ingresso Local:3:I e di configurazione Local:3:C di seguito.

Name	Alias For	Base Tag	Data Type	Style
Local:3:I			AB:1756_DI:I:0	
Local:3:I.Fault			DINT	Binary
Local:3:I.Data			DINT	Binary
Local:3:I.Data.0			BOOL	Decimal
Local:3:I.Data.1			BOOL	Decimal
Local:3:I.Data.2			BOOL	Decimal

È possibile creare un nuovo tag alias con un nome più descrittivo. Ad esempio, è possibile creare un alias per il primo ingresso denominato Limit_Switch_1, che descrive fisicamente l'ingresso.

In STEP 7, lo strumento di configurazione hardware assegnerà gli indirizzi a una scheda I/O quando viene aggiunta al sistema. Ad esempio, a una scheda di ingresso digitale possono essere assegnati i byte I16 e I17. Il programmatore identifica quindi l'indirizzo bit per ciascun ingresso ed immette un nome appropriato nella tabella dei simboli. Dopo avere eseguito questa operazione, nel programma viene creata automaticamente l'associazione I16.5 = “ZSC2036”.

Linguaggi di programmazione

In questa sezione sono descritti i linguaggi di programmazione disponibili in STEP 7 e nel software RSLogix 5000. Tutti i linguaggi non sono standard, e variano in base alla versione del software acquistato. Se si seleziona il linguaggio Logix più adatto al task, il progetto di programmazione risulterà più semplice, la codifica più rapida e il programma sarà più facile da comprendere.

Esiste una differenza significativa tra i linguaggi S7 e Logix. In S7, la lista istruzioni è il linguaggio "nativo" del controllore. Gli altri linguaggi sono trasferiti in STL. In Logix, tutti i linguaggi sono "nativi" nel controllore, e ciascun linguaggio è compilato senza fare riferimento agli altri. Il vantaggio di quest'ultimo è che quando si carica un programma dal controllore, viene visualizzato nel linguaggio in cui è stato scritto.

STEP 7 dispone di tre linguaggi standard:

- Lista istruzioni (STL): può essere descritto come assembler di alto livello
- Logica ladder (LAD)
- Diagramma a blocchi funzione (FBD)

E alcuni linguaggi opzionali:

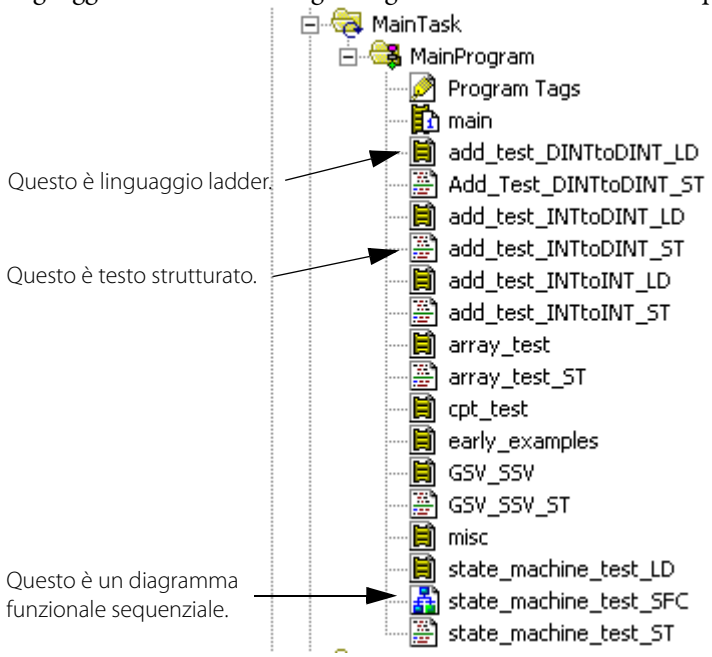
- Testo strutturato (ST)
- CFC: diagramma di flusso continuo per le applicazioni di processo
- HiGraph: controllo sequenziale tramite software di visualizzazione grafica
- ML – Motion Language: simile a GML nella versione precedente del controllore di movimento 1394 dedicato di Rockwell Automation

Un programma può essere costituito da blocchi funzione e da funzioni scritti in linguaggi diversi.

Il software RSLogix 5000 include quattro linguaggi di programmazione:

- Linguaggio ladder (LD): simile a LD Siemens, con un set di istruzioni esteso.
- Testo strutturato (ST): equivalente a ST Siemens
- Diagramma a blocchi funzione (FBD): equivalente a CFC Siemens
- Diagramma funzionale sequenziale (SFC): simile a hiGraph Siemens.

Una routine, la sezione base del codice in Logix, può essere in uno qualsiasi di questi linguaggi, e un programma può essere costituito da routine scritte in linguaggi diversi. Nell'immagine seguente viene fornito un esempio.



Linguaggio ladder Logix

Tradizionalmente, il linguaggio ladder è utilizzato per implementare la logica combinatoria booleana. In Logix, può essere inoltre utilizzato per la logica sequenziale, il controllo assi, la manipolazione dei dati e i calcoli matematici, sebbene altri linguaggi siano più adatti per tali task.

Testo strutturato Logix

Il testo strutturato è un linguaggio procedurale di alto livello di facile apprendimento per gli utenti esperti di linguaggi Basic, Pascal o uno dei linguaggi della famiglia 'C'. È utilizzato principalmente per la manipolazione dei dati e i calcoli matematici, sebbene la logica di controllo assi, combinatoria e sequenziale possa essere programmata facilmente in ST.

Diagramma a blocchi funzione Logix

Il diagramma a blocchi funzione descrive graficamente una funzione (booleana o matematica) relativa alle variabili degli ingressi e delle uscite. Le variabili degli ingressi e delle uscite sono collegate ai blocchi tramite linee di collegamento. Un'uscita di un blocco può inoltre essere collegata a un ingresso di un altro blocco.

È consigliabile programmare gli anelli PID in FBD. È il linguaggio più pratico per il controllo di processo.

Diagramma funzionale sequenziale Logix

SFC è uno strumento grafico per descrivere la logica sequenziale come un set di stati e transizioni. È possibile assegnare le uscite a uno stato e le condizioni booleane per le transizioni ad altri stati definiti.

Conversione del codice STEP 7 in Logix

- Se si desidera convertire il codice di logica ladder di STEP 7 in Logix, LD è la soluzione consigliata. Il significato di LD è simile in entrambi i sistemi.
- Se si desidera convertire il codice del diagramma a blocchi funzione di STEP 7 in Logix, FBD è la soluzione consigliata.
- Si noti che FBD Logix standard è più avanzato di FBD STEP 7, ed equivale al linguaggio CFC opzionale di STEP 7.
- Se si desidera convertire il codice della lista istruzioni di STEP 7 in Logix, il linguaggio più adatto varia in base alla natura del blocco STL. Se il blocco STL contiene soprattutto valutazioni booleane, LD è probabilmente il linguaggio Logix migliore in cui eseguire la conversione. Se il blocco STL contiene puntatori per accedere e manipolare i dati o eseguire calcoli matematici, ST è probabilmente il miglior linguaggio Logix in cui eseguire la conversione. Se il blocco STL contiene logica sequenziale, è necessario prendere in considerazione SFC sebbene la logica sequenziale possa anche essere facilmente implementata in ST e LD.

Array non puntatori

In STEP 7, gli array possono essere definiti esattamente come in Pascal o C, tuttavia i linguaggi di base (STL, LD e FBD) non dispongono di supporto di alto livello per accedervi. Al contrario, le routine puntatore devono essere costruite.

Le funzioni della libreria di STEP 7 sono prive di supporto per l'accesso agli array. I programmatori esperti di puntatori possono scrivere le proprie funzioni come FC101 “INDEXED_COPY” (vedere di seguito), tuttavia questa operazione richiede tempo e competenze.

“INDEXED_COPY” in STEP 7 ha la stessa funzione dell'istruzione COP di Logix per la copia indicizzata.

```
CALL  "INDEXED_COPY"                FC101
 indexSrc:=#index_in
 source  :="Instance_FB2".table     P#DB4.DBX0.0
 indexDst:=1
 dest    :="Instance_FB2".target    P#DB4.DBX96.0
 len     :=8
```

FC111 di seguito accederà a un array.

```
CALL  "GET_INDEXED_REFERENCE"       FC111
 refArray  :="Instance_FB2".table   P#DB4.DBX0.0
 index     :=#index_in
 byteIncr  :=32
 startIndex:=TRUE
 retVal    :=#ptr
```

Il puntatore all'oggetto viene restituito nel parametro #ptr, che può quindi essere dereferenziato per ottenere i dati.

In Logix, è possibile definire e accedere agli array con il metodo abituale utilizzato per il linguaggio di un computer di alto livello, come illustrato nel frammento riportato di seguito.

```
// copy a string from a table of strings #table
// to a target string #target. The index is #index_in

COP(table[index_in], target, target.LEN);
```

Istruzioni add-on

Panoramica delle istruzioni add-on

Le istruzioni add-on sono l'equivalente dei blocchi funzione di STEP 7, con dati riservati e scelte avanzante dei parametri. In particolare, il tipo di parametro INOUT o “passaggio per riferimento” consente di passare efficacemente le strutture di dati al codice.

Poiché l'istruzione add-on è molto simile al blocco funzione di STEP 7, è probabile che un programmatore S7 che esegue la conversione in Logix riesca a utilizzarla senza problemi.

Confronto tra FB e istruzioni add-on:

- Entrambi possono essere richiamati come funzioni con nome da qualsiasi parte del programma.
- Entrambi contengono un'area dati riservati di dati statici, sebbene non siano veramente riservati nel caso di STEP 7.
- Un blocco funzione STEP 7 include anche un'area dati temporanei.
- Nell'istruzione add-on, i dati statici locali hanno la stessa funzione.

Entrambi dispongono di tre tipi di parametri: di ingresso (passaggio per valore), di uscita (passaggio per valore) e di ingresso-uscita (passaggio per riferimento). Il parametro passaggio per riferimento rappresenta un vantaggio notevole, poiché consente di passare efficacemente strutture di dati di grandi dimensioni.

L'istruzione add-on conserva automaticamente una cronologia di modifiche mediante una registrazione cronologica e il nome utente di Windows al momento della modifica. Questa opzione non è disponibile con i blocchi funzione STEP 7.

Con l'istruzione add-on, è possibile configurare l'esecuzione di una routine di pre-scansione quando il controllore passa dalla modalità Programmazione alla modalità Esecuzione o viene attivato in modalità Esecuzione. In tali condizioni, la routine di pre-scansione verrà eseguita una volta e, in genere, può essere utilizzata per inizializzare i dati. In STEP 7 il blocco organizzativo OB100 funziona allo stesso modo, tuttavia il codice di pre-scansione non può essere collegato specificatamente a FB.

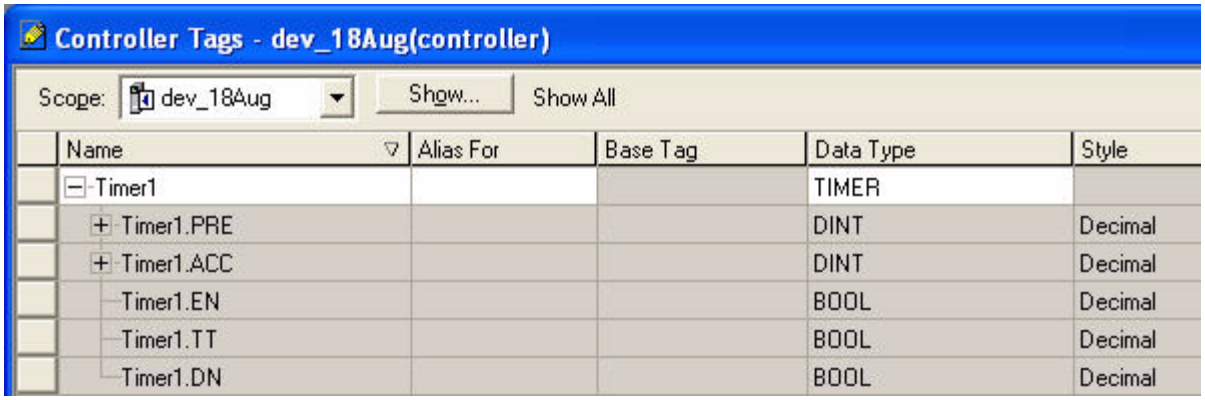
Se l'istruzione add-on viene chiamata da un passo SFC, e SFC è configurato per il ripristino automatico, una routine di post-scansione definita nell'istruzione add-on verrà eseguita una volta quando SFC termina questo passo. Può essere utilizzata per ripristinare i dati. Un FB STEP 7 non dispone di un equivalente incorporato (sebbene sia facilmente programmabile).

Un'istruzione add-on può includere una routine EnableInFalse, che sarà chiamata (se presente) quando la condizione del ramo per la chiamata all'istruzione add-on è falsa. In tal caso, i parametri di ingresso e di uscita passeranno i valori. Un FB STEP 7 non ha equivalenti.

Le istruzioni add-on sono descritte in modo più dettagliato nel Capitolo 4.

Tag di supporto

Molte istruzioni e tipi di dati utilizzano tag di supporto, ovvero tag creati specificatamente per l'istanza dell'istruzione o per i tipi di dati che si utilizzano. Le istruzioni add-on, i timer, i contatori, i messaggi e il controllo PID utilizzano tutti tag di supporto. Il software RSLogix 5000 genera la struttura di elementi corrispondente ogniqualvolta si crea un tag del tipo specifico, pertanto non è necessario creare gli elementi autonomamente.



Common Industrial Protocol (CIP)

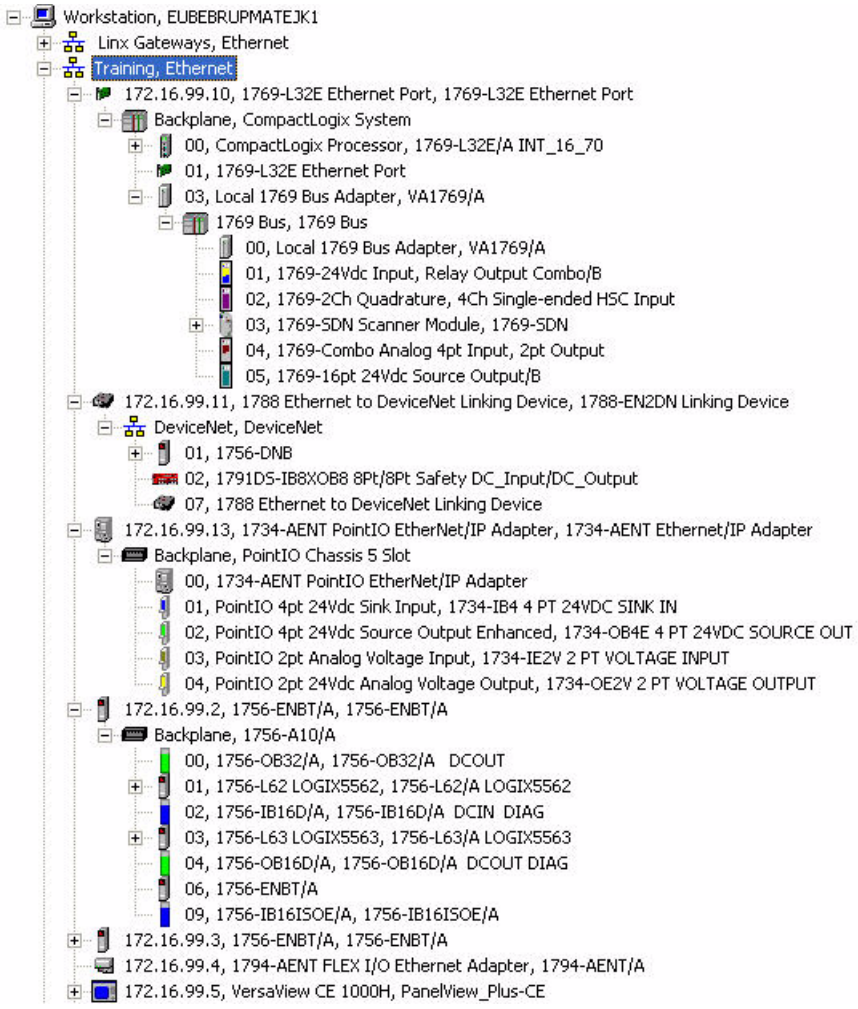
Logix utilizza tre reti principali: Ethernet/IP, ControlNet e DeviceNet. Ciascuna di esse presenta caratteristiche adatte alle diverse aree dell'applicazione. I tre tipi di rete condividono un protocollo, il 'Common Industrial Protocol' (CIP).

Il CIP consente di trasferire i dati tramite uno dei tre tipi di rete supportati da Logix con un'interfaccia di configurazione e programmazione praticamente identica per tutti e tre. Inoltre, i dati possono essere trasferiti mediante una rete creata da più di uno dei tre tipi di rete senza che il programmatore debba tradurre i protocolli.

Nell'S7 "tradizionale" i due protocolli principali sono Industrial Ethernet, per il collegamento al sistema IT e ad altri controllori, e Profibus DP per il collegamento ai sistemi di campo. Questi due protocolli sono separati a livello di hardware e a livello di dati. Con l'hardware e il software S7 più recenti, "Profinet CBA" integra Industrial Ethernet, Profinet e Profibus.

Visualizzazione della rete

Gli utenti S7 possono trovare la configurazione e la gestione della rete Logix impressionante. La struttura ad albero di esempio riportata di seguito mostra i dispositivi effettivamente collegati al sistema. Questa struttura è stata generata andando online e non è stato configurato alcun elemento.


Workstation, EUBEBRUPMATEJK1
Linx Gateways, Ethernet
Training, Ethernet
172.16.99.10, 1769-L32E Ethernet Port, 1769-L32E Ethernet Port
Backplane, CompactLogix System
00, CompactLogix Processor, 1769-L32E/A INT_16_70
01, 1769-L32E Ethernet Port
03, Local 1769 Bus Adapter, VA1769/A
1769 Bus, 1769 Bus
00, Local 1769 Bus Adapter, VA1769/A
01, 1769-24Vdc Input, Relay Output Combo/B
02, 1769-2Ch Quadrature, 4Ch Single-ended HSC Input
03, 1769-SDN Scanner Module, 1769-SDN
04, 1769-Combo Analog 4pt Input, 2pt Output
05, 1769-16pt 24Vdc Source Output/B
172.16.99.11, 1788 Ethernet to DeviceNet Linking Device, 1788-EN2DN Linking Device
DeviceNet, DeviceNet
01, 1756-DNB
02, 1791DS-IB8XOB8 8Pt/8Pt Safety DC_Input/DC_Output
07, 1788 Ethernet to DeviceNet Linking Device
172.16.99.13, 1734-AENT PointIO EtherNet/IP Adapter, 1734-AENT Ethernet/IP Adapter
Backplane, PointIO Chassis 5 Slot
00, 1734-AENT PointIO EtherNet/IP Adapter
01, PointIO 4pt 24Vdc Sink Input, 1734-IB4 4 PT 24VDC SINK IN
02, PointIO 4pt 24Vdc Source Output Enhanced, 1734-OB4E 4 PT 24VDC SOURCE OUT
03, PointIO 2pt Analog Voltage Input, 1734-IE2V 2 PT VOLTAGE INPUT
04, PointIO 2pt 24Vdc Analog Voltage Output, 1734-OE2V 2 PT VOLTAGE OUTPUT
172.16.99.2, 1756-ENBT/A, 1756-ENBT/A
Backplane, 1756-A10/A
00, 1756-OB32/A, 1756-OB32/A DCOUT
01, 1756-L62 LOGIX5562, 1756-L62/A LOGIX5562
02, 1756-IB16D/A, 1756-IB16D/A DCIN DIAG
03, 1756-L63 LOGIX5563, 1756-L63/A LOGIX5563
04, 1756-OB16D/A, 1756-OB16D/A DCOUT DIAG
06, 1756-ENBT/A
09, 1756-IB16ISOE/A, 1756-IB16ISOE/A
172.16.99.3, 1756-ENBT/A, 1756-ENBT/A
172.16.99.4, 1794-AENT FLEX I/O Ethernet Adapter, 1794-AENT/A
172.16.99.5, VersaView CE 1000H, PanelView_Plus-CE

Le reti sono descritte in modo più dettagliato nel Capitolo 1.

Scambio di dati tra controllori

Invio/ricezione in STEP 7

Per preparare la comunicazione tra controllori in STEP 7, vengono eseguiti i seguenti passaggi.

1. Le stazioni remote sono configurate graficamente in un componente STEP 7 denominato NetPro.
2. In NetPro viene creata una tabella di collegamenti, in cui sono specificati i protocolli e i parametri per ciascun collegamento.
3. Le funzioni della libreria FC5 AG_SEND e FC6 AG_RECV sono copiate nel progetto.
4. Le chiamate vengono eseguite dal programma utente a AG_SEND e AG_RECV, specificando i parametri dei collegamenti e le aree di dati utilizzate per fornire e ricevere i dati.

Tag prodotti/consumati in Logix

I tag prodotti e consumati rappresentano il mezzo per trasferire i dati critici tra i controllori Logix collegati in rete a intervalli di tempo definiti. I tag prodotti e consumati possono essere trasmessi su Ethernet/IP o ControlNet e sul backplane dei controllori ControlLogix.

I tag prodotti e consumati sono tag configurati come prodotti o consumati al momento della creazione. Se un tag è contrassegnato come prodotto, il relativo valore sarà indirizzato globalmente a una rete EtherNet/IP o ControlNet alla quale è collegato il controllore. Se è contrassegnato come consumato, il controllore al quale il tag richiede i dati sarà identificato come parte della configurazione e il tag consumato riceverà il proprio valore dal tag prodotto equivalente in tale controllore.

Per l'invio e la ricezione sono presenti canali separati. La modifica del valore di un tag consumato non influirà sul tag prodotto. Questa comunicazione è simile a quella tra controllori in S7 e si differenza dalla comunicazione tra controllore e SCADA, in cui le modifiche vengono riflesse dalla controparte.

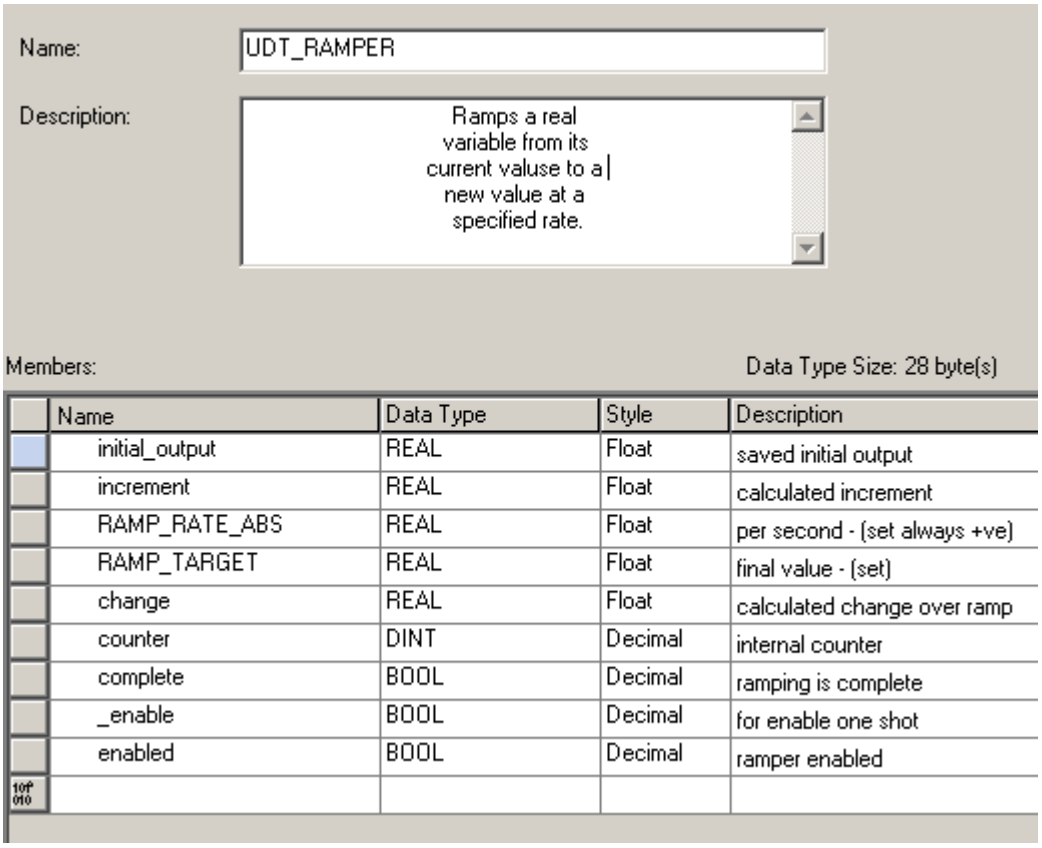
Per configurare i collegamenti del sistema produttore/consumatore non è necessaria programmazione, mentre in S7 è necessario programmare la comunicazione tra controllori (SEND/RECEIVE).

Tipi di dati definiti dall'utente

In Logix, è possibile configurare dei tipi di dati definiti dall'utente. In questo modo, è possibile dichiarare la struttura di un tipo di dati complesso come tipo. Le istanze di tale tipo possono quindi essere definite nel programma.

I tipi di dati definiti dall'utente di Logix sono configurati e utilizzati in modo simile a quelli di STEP 7.

UDT Logix


Name:
UDT_RAMPER
Description:
Ramps a real variable from its current valuse to a new value at a specified rate.
Members:
Data Type Size: 28 byte(s)

Name	Data Type	Style	Description
initial_output	REAL	Float	saved initial output
increment	REAL	Float	calculated increment
RAMP_RATE_ABS	REAL	Float	per second - (set always +ve)
RAMP_TARGET	REAL	Float	final value - (set)
change	REAL	Float	calculated change over ramp
counter	DINT	Decimal	internal counter
complete	BOOL	Decimal	ramping is complete
_enable	BOOL	Decimal	for enable one shot
enabled	BOOL	Decimal	ramper enabled

Aggiornamento asincrono degli I/O

Nei sistemi Logix, l'I/O viene aggiornato in modo asincrono rispetto ai periodi di esecuzione del programma, a differenza dei PLC tradizionali come S7, in cui una tabella immagine I/O viene aggiornata all'inizio del ciclo e i valori degli ingressi non variano durante l'esecuzione del programma.

Il programmatore Logix deve valutare se è necessario un buffer dei dati di ingresso, in modo che il loro valore rimanga costante durante l'esecuzione del programma.

È abbastanza comune “consumare” gli ingressi solo una volta passandoli come parametri a un modulo del codice. Gli ingressi non saranno utilizzati in nessuna altra parte del programma, pertanto non è necessario eseguire il buffering. Vedere l'esempio di modulo di controllo nel Capitolo 4.

Tipo di dati DINT

I controllori Logix funzionano con tag DINT (numero intero a 32 bit) in modo più efficiente che con INT (numero intero a 16 bit) o SINT (numero intero a 8 bit). Utilizzare DINT ove possibile, anche se l'intervallo di valori utilizzato si adatterebbe a un tag INT o SINT. Questi tipi di dati sono forniti per motivi di compatibilità con IEC61131-3, tuttavia sono convertiti internamente in DINT prima di essere utilizzati dal programma, in modo che il codice venga eseguito più efficacemente nella maggior parte delle situazioni.

Phase Manager

Gestione delle fasi in STEP 7

STEP 7 non dispone di strumenti incorporati per eseguire la gestione delle fasi. Le strutture necessarie devono essere programmate in un set di routine, in genere chiamato PLI o Phase Logic Interface. Di seguito sono riportati i componenti di un programma PLI basato su S88.

- Un sequenziatore di passi il cui funzionamento è conforme al modello a stati S88. Alcuni passi o intervalli di passi definiscono lo stato S88. I comandi del sequenziatore sono inoltre specificati da S88, e il sequenziatore risponde solo quando lo consente il modello a stati. Un sequenziatore con tali proprietà è denominato fase.

- Un set di dati per ciascuna fase utilizzato per registrare lo stato della fase e per ricevere i comandi in entrata dal gestore di ricette. Il gestore di ricette comunica con questi dati. Il formato dei dati varia in base al gestore di ricette.

- Un modulo logico che traduce lo stato della fase in un formato richiesto dal gestore di ricette e traduce i comandi dal gestore di ricette in comandi della fase.

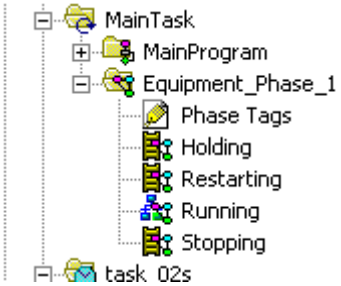
PhaseManager in Logix

In una fase dell'apparecchiatura S88, vi sono stati specifici della fase nonché transizioni tra gli stati. PhaseManager è una funzionalità del software RSLogix 5000 che consente di eseguire tre operazioni:

- assegnare il codice per ciascuno stato della fase a una routine diversa
- eseguire automaticamente una macchina a stati in grado di gestire le transizioni tra gli stati della fase
- gestire l'esecuzione della fase utilizzando un set di comandi Logix.

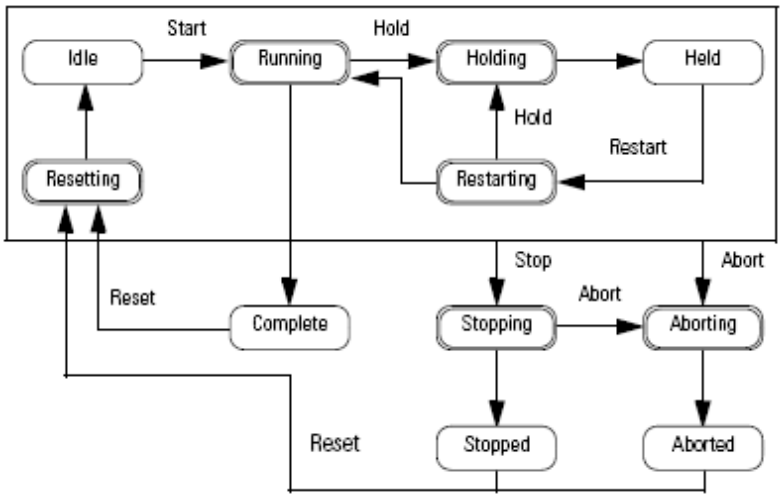
È utilizzato in svariate aree dell'applicazione, inclusi, ad esempio, il controllo di processo e l'imballaggio, poiché consente una netta separazione di controllo dispositivo/apparecchiature e controllo procedurale, rendendo quindi il codice molto più modularizzato ed efficiente, specialmente per sistemi di grandi dimensioni.

Fase dell'apparecchiatura nella struttura ad albero del progetto



Il codice per ciascuno stato della fase può essere scritto in uno qualsiasi dei linguaggi Logix.

Si tratta della macchina a stati di fase. È quasi identica al modello a stati S88.



Se è stata programmata una routine Phase Manager/PLI STEP 7 compatibile con S88 e si desidera convertirla in Logix, può essere possibile evitare la traduzione utilizzando PhaseManager di Logix.

Tempo di sistema coordinato (CST)

S7 dispone di un orologio di sistema, rappresentato utilizzando 32 bit e impulsi in millisecondi. È possibile ottenerne (e memorizzarne) il valore eseguendo una chiamata al sistema operativo, utile per la misurazione precisa degli intervalli di tempo.

Logix utilizza il tempo di sistema coordinato che è un numero a 64 bit che misura il numero di microsecondi dall'ultimo avvio del controllore. Come con S7, gli intervalli possono essere misurati eseguendo chiamate al sistema operativo per ottenere il valore CST. Rappresenta la base per la sincronizzazione dell'orologio per sistemi a più CPU, funzionalità di controllo assi precise, commutazione delle uscite schedulata precisa a 100 µs, registrazione cronologica degli eventi degli ingressi, campionamento analogico schedulato, monitoraggio e comunicazione dei Safety I/O, calcoli della posizione della camma e Wall Clock Time.

Ingressi con registrazione cronologica

La registrazione cronologica è una funzionalità che consente di registrare una modifica dei dati di ingresso con ~~un~~ tempo relativo del momento in cui si è verificata la modifica. Con i moduli di ingresso digitali, è possibile configurare una registrazione cronologica per le modifiche dei dati. È possibile utilizzare la registrazione cronologica CST per confrontare il tempo relativo tra i campioni di dati.

Ciò consente al programmatore di ottenere una precisione impareggiabile nel collegamento dei segnali di ingresso ai riferimenti temporali per applicazioni, come quelle utilizzate frequentemente nel controllo assi, senza sovraccaricare i sistemi di comunicazione e di elaborazione della logica e il codice applicativo correlato.

Uscite schedulate

Con i moduli di uscita digitali, è possibile configurare il modulo per impostare le uscite a un'ora schedulata.

Ciò consente al programmatore di ottenere una precisione impareggiabile nel collegamento delle uscite ai riferimenti temporali per applicazioni, come le posizioni dell'asse nel controllo assi, o le funzioni di controllo di processo, senza sovraccaricare i sistemi di comunicazione e di elaborazione della logica e il codice applicativo correlato.

Nessuna variabile temporanea

S7 include una categoria di variabili denominate variabili temporanee. Il relativo ambito è il blocco di programmazione nel quale sono definite e il relativo ciclo di vita è l'esecuzione del blocco di programmazione nel quale sono definite.

Logix non dispone di un equivalente della variabile temporanea. Tutte le variabili sono statiche, e conservano i propri valori finché non vengono modificate.

Per ottenere le funzionalità in genere richieste nelle applicazioni S7, adottare ad esempio uno dei seguenti approcci:

- utilizzare tag del programma
- se si programma un'istruzione add-on, utilizzare i tag locali (parte dei dati dell'istruzione add-on).

Non sono necessari accumulatori o registri speciali

Se si programma nella lista di istruzioni di STEP 7, si avrà esperienza con gli accumulatori e i registri puntatore AR1 e AR2. Non esistono equivalenti in Logix. Tutti gli operandi sono tag.

Per ottenere le funzionalità in genere richieste nelle applicazioni S7, adottare ad esempio uno dei seguenti approcci:

- utilizzare tag del programma
- se si programma un'istruzione add-on, utilizzare i tag locali (parte dei dati dell'istruzione add-on)
- valutare se sono necessari equivalenti Logix degli accumulatori S7 e registri speciali. Sono presenti per la natura di basso livello della lista istruzioni S7 e, in un linguaggio come il testo strutturato, è probabile che non siano necessari.

Conversione del software di sistema e delle funzioni standard

Introduzione

In questo capitolo sono elencate le funzioni del sistema S7 utilizzate più frequentemente, viene illustrato come sono eseguite le operazioni equivalenti in Logix e sono forniti diversi esempi specifici.

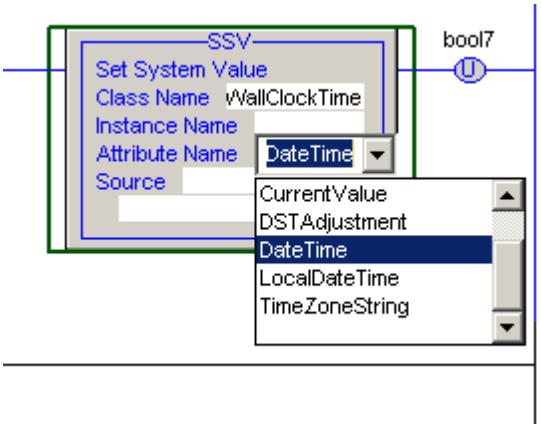
Argomento	Pagina
Funzioni del sistema Logix	68
Copia	68
Impostazione o lettura di data e ora	69
Lettura dell'ora del sistema	69
Gestione degli interrupt	70
Errori	70
Stato – Controllore	71
Stato – Modulo	71
Stato – per OB e task	72
Timer	72
Routine di conversione	73
Routine di gestione delle stringhe	73
Esempi di chiamate di funzione al sistema	74

Lo scopo di questo capitolo è di informare l'utente sulle istruzioni dedicate disponibili in Logix, in modo da non perdere tempo a sviluppare soluzioni che esistono già.

Funzioni del sistema Logix

In Logix, l'equivalente della maggior parte delle funzioni del sistema S7 sono le istruzioni GSV (Get System Value) e SSV (Set System Value). Queste istruzioni permettono di accedere ad una gerarchia di oggetti (classi, istanze e attributi) incorporata nei controllori Logix. Se si programmano GSV e SSV, sarà possibile selezionare i parametri tramite menu a discesa.

Istruzione SSV



Una volta appresi i principi di base di GSV e SSV, il nuovo utente di Logix potrebbe rendersi conto che l'accesso al sistema operativo risulta più facile che con gli SFC di S7.

Copia

Utilizzata per copiare strutture di dati complesse, array di istanze dei tipi di dati utente.

S7	Commento	Logix	Commento
SFC20 BLKMOV	Con BLKMOV, gli indirizzi devono essere definiti al momento della compilazione.	COP (istruzione)	Se COP viene utilizzato per eseguire la copia tra array, l'inizio del blocco (origine o destinazione) può includere un indice di array relativo all'elemento il cui valore è valutato al momento del runtime.
SFC81 UBLKMOV	Versione non interrompibile: per garantire che i dati di origine non possano essere modificati durante la copia.	CPS (istruzione)	Versione non interrompibile: per garantire che i dati di origine non possano essere modificati durante la copia.
SFC14 DPRD_DAT	Se il dispositivo Profibus DP dispone dell'area dati di comunicazione > 4 byte, SFC garantisce valori di lettura coerenti.	CPS (ControlNet e Ethernet /IP)	Non necessario per DeviceNet
SFC15 DPWR_DAT	Se il dispositivo Profibus DP dispone dell'area dati di comunicazione > 4 byte, SFC garantisce valori di scrittura coerenti.	CPS (ControlNet e Ethernet /IP)	Non necessario per DeviceNet

Impostazione o lettura di data e ora

I controllori di entrambi i sistemi dispongono di un orologio in tempo reale, che può essere letto o impostato.

S7	Commento	Logix	Commento
SFC0 SET_CLK	Valori passati in un'istanza del tipo DT (DateTime)	SSV (Set System Value)	Classe SSV - WallClockTime Attributo SSV - DateTime Origine SSV - specifica l'elemento[0] di DINT[7]
SFC1 READ_CLK	Valori restituiti in un'istanza del tipo DT (DateTime)	GSV (Get System Value)	Classe GSV - WallClockTime Attributo GSV - DateTime Dest. GSV – elemento[0] di DINT[7]

Lettura dell'ora del sistema

I controllori di entrambi i sistemi dispongono di un orologio di sistema, che inizia quando viene avviato il controllore. Nel sistema S7, l'ora è in millisecondi, in Logix in microsecondi.

S7	Commento	Logix	Commento
SFC64 TIME_TCK	Restituisce l'ora del sistema nell'intervallo 0…2,31 ms	GSV (Get System Value)	Restituisce l'ora del sistema nell'intervallo 0…2,63 µs Classe GSV - CST Attributo GSV - CurrentValue Dest. GSV - specifica l'elemento[0] di DINT[2] DINT[0] - 32 bit più bassi DINT[1] - 32 bit più alti

Gestione degli interrupt

Gli interrupt possono essere abilitati e disabilitati dal programma utente che esegue chiamate alle funzioni di sistema.

S7	Commento	Logix	Commento
SFC39 DIS_IRT	Disabilita gli interrupt gestiti da un OB specifico. Le richieste di interrupt vanno perse.	SSV Inibisce il task specificato.	Classe SSV - Task Istanza SSV - Nome task Attributo SSV - InhibitTask Origine SSV - Variabile DINT impostata su 1
SFC39 EN_IRT	Abilita gli interrupt gestiti da un OB specifico.	SSV Abilita un task specifico.	Classe SSV - Task Istanza SSV - Nome task Attributo SSV - InhibitTask Origine SSV - Variabile DINT impostata su 0
SFC41 DIS_AIRT	Disabilita gli interrupt gestiti da un OB specifico. Le richieste di interrupt sono ritardate.	UID	Disabilita l'interruzione del task corrente da un task a priorità più alta
SFC42 EN_AIRT	Abilita gli interrupt gestiti da un OB specifico. Gli interrupt ritardati da SFC41 vengono eseguiti.	UIE	Abilita gli interrupt del task corrente.

Errori

Queste chiamate al sistema restituiscono campi di bit nel caso di S7, o un numero intero nel caso di Logix, che rappresentano i codici di errore.

S7	Commento	Logix	Commento
SFC38 READ_ERR	Legge e cancella i bit di errore. Il tipo di errore da interrogare può essere selezionato con il campo di filtraggio.	GSV (Utilizzare SSV per ripristinare contatori o errori)	Classe GSV - FaultLog Attributo GSV: MajorEvents – Nessun evento principale MinorEvents – Nessun evento secondario MajorFaultBits – Errore grave corrente MinorFaultBits – Errore minore corrente Destinazione GSV – INT o DINT per ricevere i dati

Stato – Controllore

La chiamata a SFC (S7) e GSV (Logix) restituisce i dati sul controllore. Nota: è necessario acquisire esperienza prima di poter utilizzare SFC51. GSV in questo caso è più accessibile.

S7	Commento	Logix	Commento
SFC51 RDSYSST	I parametri di ingresso specificano la classe di informazioni da leggere e, se possibile, un numero di istanza se sono presenti più oggetti. I parametri di uscita sono un puntatore a un elenco con le informazioni restituite, nonché il numero e la dimensione degli elementi nell'elenco.	GSV	Moduli con un collegamento diretto: esaminare il membro 'Fault' o 'ChannelFault', se presente. Moduli con collegamento ottimizzato per rack: esaminare il membro 'SlotStatusBits' dei dati di ingresso della scheda di comunicazione o il membro 'Fault' della scheda, come sopra. Per tutte le altre schede: eseguire GSV: Classe – Modulo Istanza – ModuleName Attributo - Entrystatus

Stato – Modulo

La chiamata a SFC (S7) e GSV (Logix) restituisce i dati sui moduli installati.

S7	Commento	Logix	Commento
SFC51 RDSYSST	I parametri di ingresso specificano la classe di informazioni da leggere e, se possibile, un numero di istanza se sono presenti più oggetti. I parametri di uscita sono un puntatore a un elenco con le informazioni restituite, nonché il numero e la dimensione degli elementi nell'elenco.	GSV	Classe GSV - Modulo Attributo GSV: EntryStatus (rapporto dell'oggetto Module con il modulo) FaultCode FaultInfo ForceStatus LEDStatus Modalità (anche SSV) Destinazione GSV – Varia in base all'attributo scelto

È possibile monitorare le informazioni sugli errori nei tag Logix creati quando il modulo viene inserito nella configurazione I/O. Analogamente, con STEP 7, se si accede alla configurazione hardware e si passa a "Open ONLINE", saranno visualizzate le informazioni per i moduli.

Stato – per OB e task

S7	Commento	Logix	Commento
Intestazione OB	I dati di stato per gli OB sono memorizzati nelle variabili temporanee generate automaticamente dall'intestazione OB. È possibile accedervi direttamente dal codice OB e trasferirli alle aree di dati statici se è necessario eseguire l'accesso dall'esterno dell'OB. Vedere un esempio di seguito.	GSV / SSV	Classe GSV - Task Istanza GSV - Nome task Attributo GSV: DisableUpdateOutputs (al termine del task) EnableTimeOut InhibitTask Istanza LastScanTime (microsecondi) MaxIntervaln (tra le esecuzioni successive del task) OverlapCount (attivato durante l'esecuzione) Priorità Rate (periodo in microsecondi) StartTime (valore di WallClockTime all'ultimo avvio del task) Stato (3 bit di stato) Watchdog (microsecondi) GSV Source / Target – Varia in base all'attributo scelto

Timer

S7	Commento	Logix	Commento
SFB4 TON	Timer con ritardo all'eccitazione	TON (LD) TONR (ST e FBD)	Timer con ritardo all'eccitazione
		RTO (LD) RTOR (LD e ST)	Timer ritentivo con ritardo all'eccitazione
SFB5 TOF	Timer con ritardo alla diseccitazione	TOF (LD) TOFR (ST e FBD)	Timer con ritardo alla diseccitazione
SFB3 TP	Genera un impulso che verrà eseguito incondizionatamente	Bit dell'accumulatore di un TON a esecuzione libera	

Routine di conversione

S7	Commento	Logix	Commento
Funzioni della libreria		**Istruzioni**	
FC16 I_STRNG	Da numero intero a stringa	DTOS	È possibile utilizzare INT come un tag di origine anziché DINT
FC5 DI_STRNG	Da numero intero doppio a stringa	DTOS	Da DINT a stringa
FC30 R_STRG	Da reale a stringa	RTOS	Da reale a stringa
FC38 STRG_I	Da stringa a numero intero	DTOS	
FC37 STRG_DI	Da stringa a numero intero doppio	STOD	Da stringa a DINT
FC39 STRG_R	Da stringa a reale	STOR	Da stringa a reale

Routine di gestione delle stringhe

S7	Commento	Logix	Commento
	Funzioni della libreria		Istruzioni
FC10 EQ_STRNG	Confronto di stringhe per uguaglianza	EQU	Confronto di stringhe per uguaglianza
FC13 GE_STRNG	Confronto di stringhe per >=	GEQ (LD) >= (ST)	Confronto di stringhe per >=
FC15 GT_STRNG	Confronto di stringhe per >	GRT (LD)	Confronto di stringhe per >
FC19 LE_STRNG	Confronto di stringhe per <=	LEQ (LD) <= (ST)	Confronto di stringhe per <=
FC24 LT_STRNG	Confronto di stringhe per <	LES (LD) < (ST)	Confronto di stringhe per <
FC29 NE_STRNG	Confronto di stringhe per <>	NEQ (LD) <> (ST)	Confronto di stringhe per <>
FC21 LEN	Lunghezza stringa	.LEN	Proprietà di qualsiasi istanza di stringa
FC26 MID	Restituisce la metà di una stringa	MID	Restituisce la metà di una stringa
FC2 CONCAT	Concatenazione di due stringhe	CONCAT	Concatenazione di due stringhe
	Può essere eseguita con FC31 REPLACE	DELETE	Eliminazione della sezione di una stringa
FC17 INSERT	Inserimento di una stringa di origine in una stringa di destinazione	INSERT	Inserimento di una stringa di origine in una stringa di destinazione
FC31 REPLACE	Sostituzione di n caratteri della stringa di destinazione con la stringa di origine	Utilizzare DELETE / INSERT	
FC11 FIND	Ricerca di una stringa in un'altra stringa	FIND	Ricerca di una stringa in un'altra stringa

In STEP 7 non esiste equivalente delle istruzioni della porta seriale ASCII di Logix, né nel set di istruzioni né nella libreria delle funzioni. Occorre pertanto programmarle in STL, se necessario.

Esempi di chiamate di funzione al sistema

Questi esempi hanno principalmente lo scopo di illustrare l'utilizzo delle istruzioni GSV/SSV.

Impostazione dell'orologio

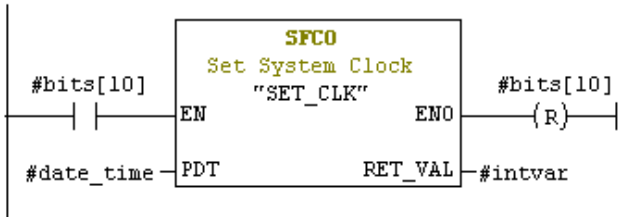
STEP 7

Questa chiamata a SFC0 imposta l'orologio. La data e l'ora vengono immesse in #date_time.

La data e l'ora sono memorizzate in 8 byte dopo #data_time in formato BCD.

Network 14 : Title:

```
set the clock to the value stored in "date_time"
```



0 – anno

1 – mese

2 – giorno

3 – ore

4 – minuti

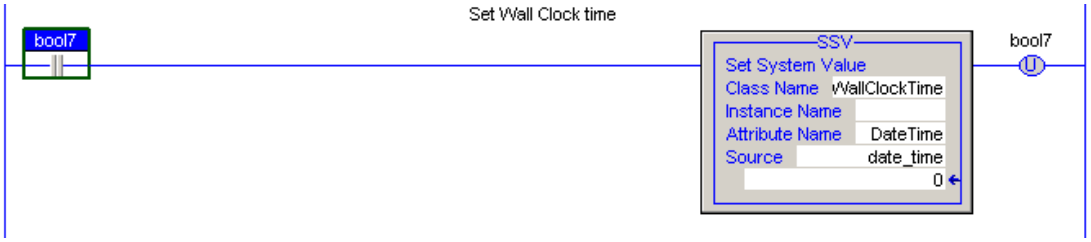
5 – secondi

6 – 2 cifre più significative di millisecondi

7 – 1 cifra meno significativa di millisecondi e giorno della settimana

Logix

I valori di data e ora sono memorizzati in sette DINT dopo #date_time.



0 – anno

1 – mese

2 – giorno

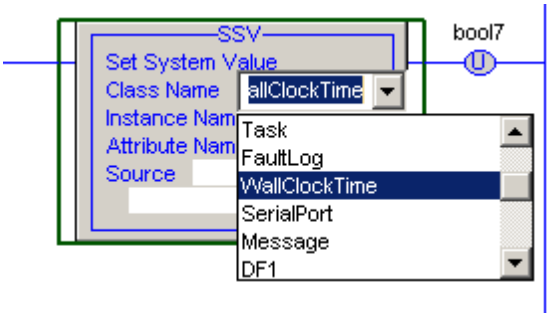
3 – ore

6 – minuti

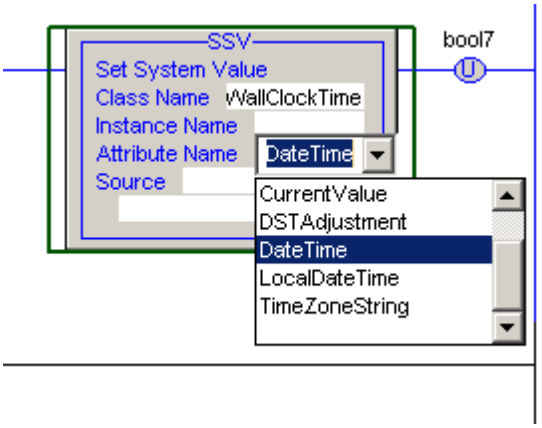
5 – secondi

6 – microsecondi

Nell'immagine relativa a Logix è mostrata la struttura di dati associata a GSV e SSV. Selezionare la classe da un menu a discesa come segue.



Selezionare l'attributo dal menu a discesa come segue.



Infine, selezionare il tag che sarà l'origine (SSV) o la destinazione (GSV) dei dati.

Disabilitazione degli interrupt

STEP 7

```
Network 2 : Title:
Disable interrupts for the Interrupt Execution (ie Periodic) OB35

          CALL  "DIS_IRT"                SFC39          -- Disable New Inte
           MODE   :=B#16#2
           OB_NR  :=35
           RET_VAL:=#intVar
```

Logix

In questo esempio è mostrato SSV nel testo strutturato.

Se si digita “gsv”, quindi “alt-A”, sarà visualizzata la schermata di selezione dei parametri.

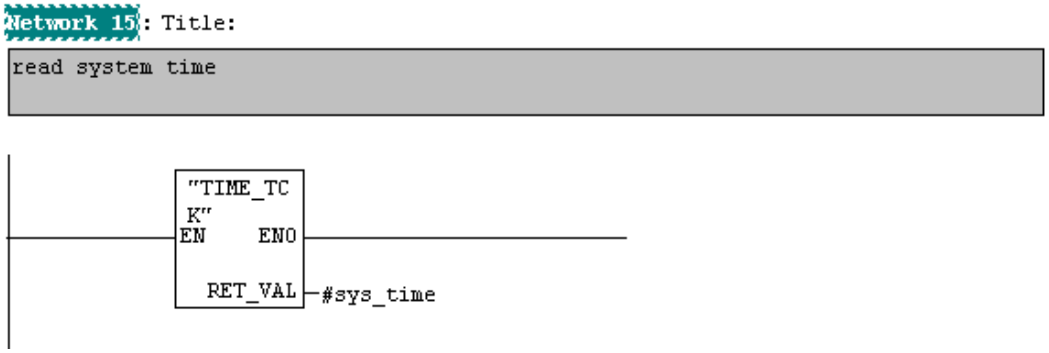
```
// disable task_0.2s
ssv()
```

Dopo avere immesso i parametri, fare clic su “OK” per completare i parametri effettivi.

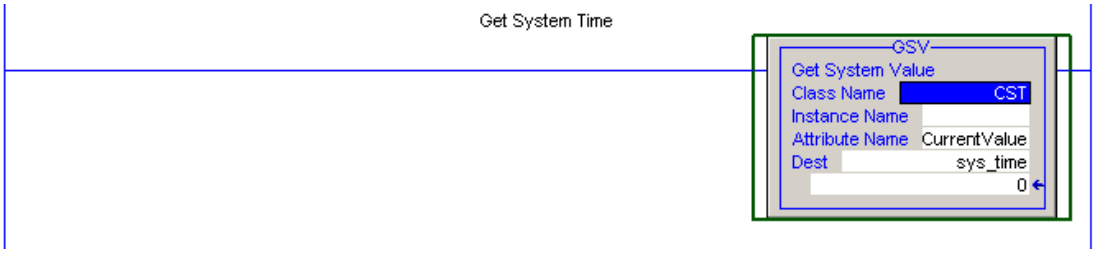
```
// disable task_0.2s
ssv(Task,task_02s,InhibitTask,disable);
```

Lettura dell'ora del sistema

STEP 7



Logix

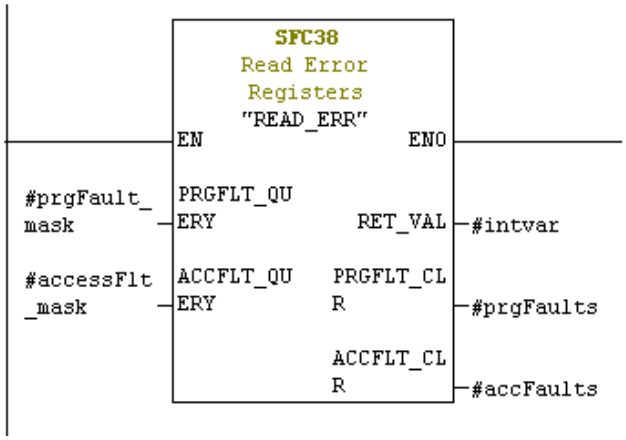


Determinazione degli errori

STEP 7

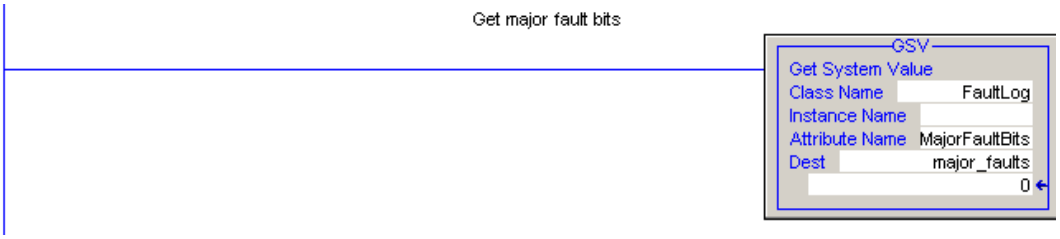
Network 16 : Title:

```
Get programming faults and I/O access faults
```



La sequenza di bit nei parametri di ingresso funge da filtro per selezionare gli errori da interrogare. Gli errori restituiti sono **mascherati**: il mascheramento evita che interrompano il controllore o chiamino un OB di errori.

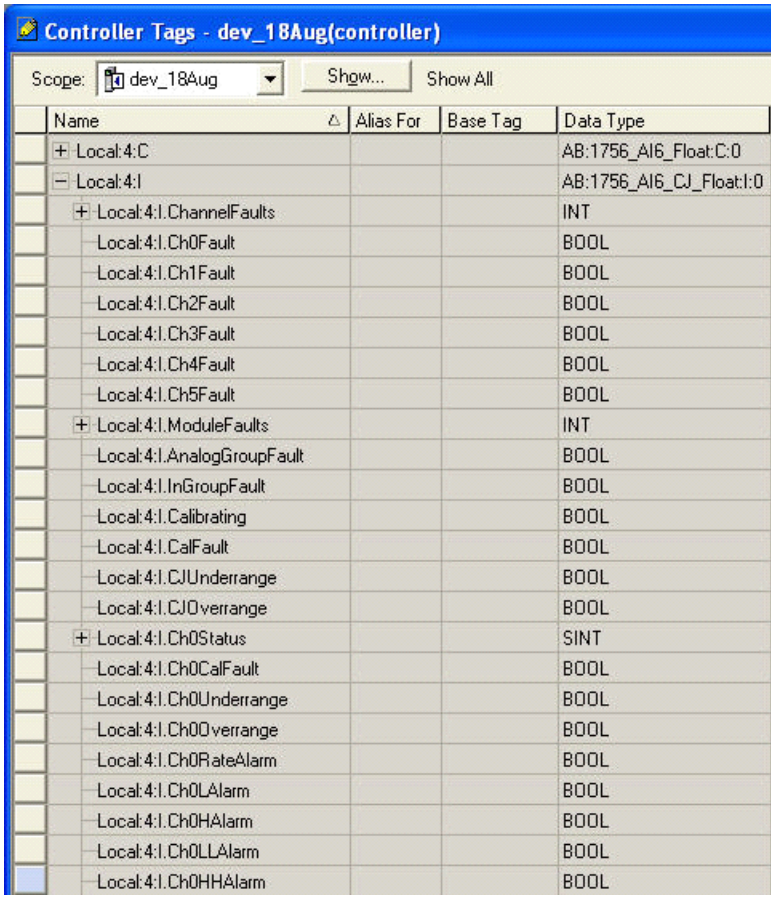
Logix



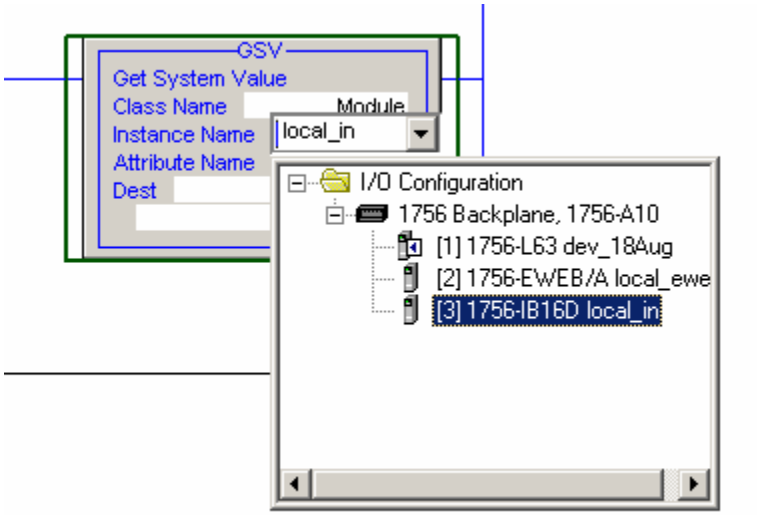
Informazioni sul modulo

Il modo più semplice è verificare i tag di profilo del dispositivo del modulo, che contengono informazioni su errori/diagnostica.

Tag della scheda di ingresso analogica termocoppia 1756-IT6I2



Un altro metodo consiste nell'utilizzare l'istruzione GSV per leggere gli oggetti del modulo. Nella seguente immagine è illustrato come utilizzare GSV per ottenere informazioni relative al modulo di ingresso digitale 1756-IB16D.



Determinazione del tempo di scansione

STEP 7

Questa immagine è relativa all'intestazione delle variabili temporanee per OB1.

Contents Of: 'Environment\Interface\TEMP'

Name	Data Type	Address	Comment
OB1_SCAN_1	Byte	1.0	1 (Cold restart scan 1 of OB 1), 3 (Scan 2-n of OB 1)
OB1_PRIORITY	Byte	2.0	Priority of OB Execution
OB1_OB_NUMBR	Byte	3.0	1 (Organization block 1, OB1)
OB1_RESERVED_1	Byte	4.0	Reserved for system
OB1_RESERVED_2	Byte	5.0	Reserved for system
OB1_PREV_CYCLE	Int	6.0	Cycle time of previous OB1 scan (milliseconds)
OB1_MIN_CYCLE	Int	8.0	Minimum cycle time of OB1 (milliseconds)
OB1_MAX_CYCLE	Int	10.0	Maximum cycle time of OB1 (milliseconds)
OB1_DATE_TIME	Date_And...	12.0	Date and time OB1 started

#OB1_PREV_CYCLE è il tempo di scansione. Come una variabile temporanea, smette di esistere quando l'esecuzione di OB1 è completa. Per memorizzare il tempo di scansione, copiare #OB1_PREV_CYCLE in una posizione della memoria statica.

Logix

È possibile recuperare il tempo di esecuzione per ciascun task Logix.

Con S7, è possibile ottenere direttamente il tempo di scansione per OB1 da #OB1_PREV_CYCLE. Tuttavia, per gli OB periodici non esiste un equivalente di #OB1_PREV_CYCLE. Per ottenere il tempo di esecuzione per gli OB periodici, è necessario inserire le chiamate a SFC64 TIME_TCK all'inizio e alla fine dell'OB e sottrarre le ore dell'orologio di sistema restituite da SFC.

Conversione delle strutture di programma tipiche

Introduzione

L'obiettivo di questa sezione è dimostrare in che modo i task di programmazione tipici in STEP 7 possono essere eseguiti nel software RSLogix 5000. L'analisi si basa soprattutto su frammenti di codice, tuttavia sono forniti anche esempi completi.

Argomento	Pagina
Esempi del codice di conversione	83
Altri argomenti relativi alla programmazione	120
Un esempio più ampio - Modulo di controllo	121

Sono inoltre inclusi argomenti relativi alla programmazione, come l'ambito e la visibilità delle variabili e la schedulazione delle sezioni di codice.

Esempi del codice di conversione

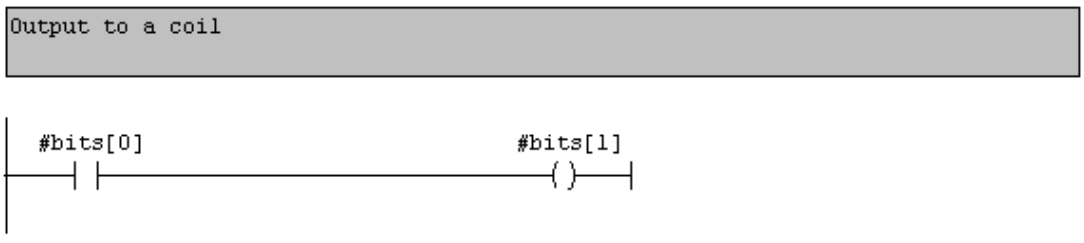
In questi esempi è illustrato il codice di conversione.

Traduzione della logica ladder

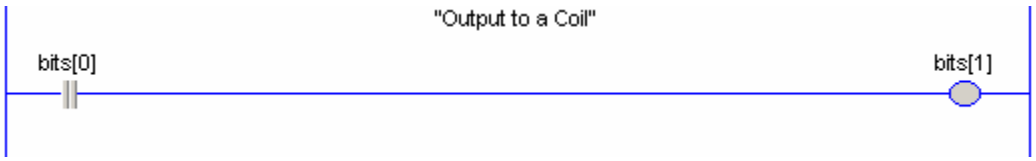
In questa sezione sono riportati alcuni esempi di confronto tra il LAD di STEP 7 e l'LD di Logix.

Scrittura su una bobina

STEP 7

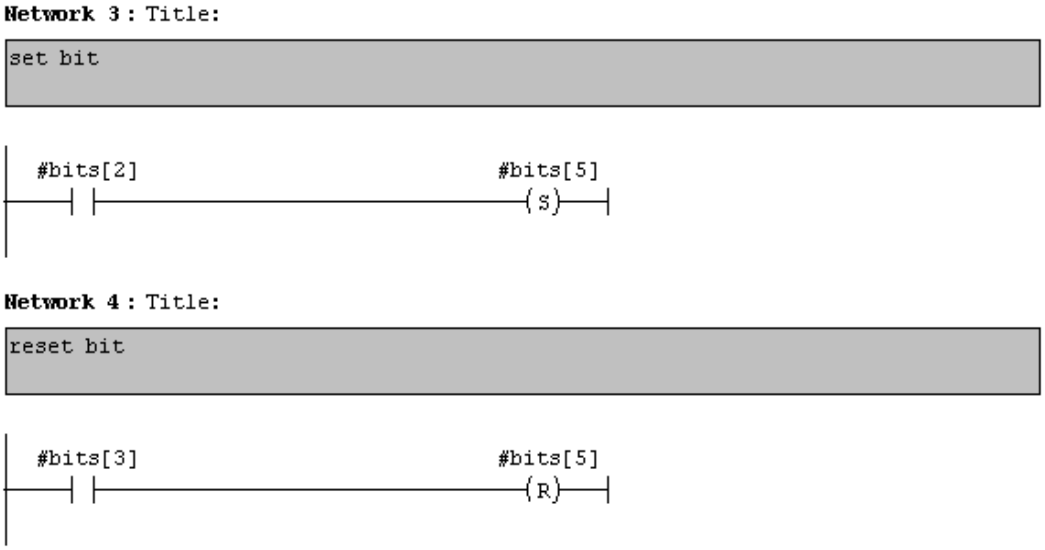


LOGIX



Set e reset

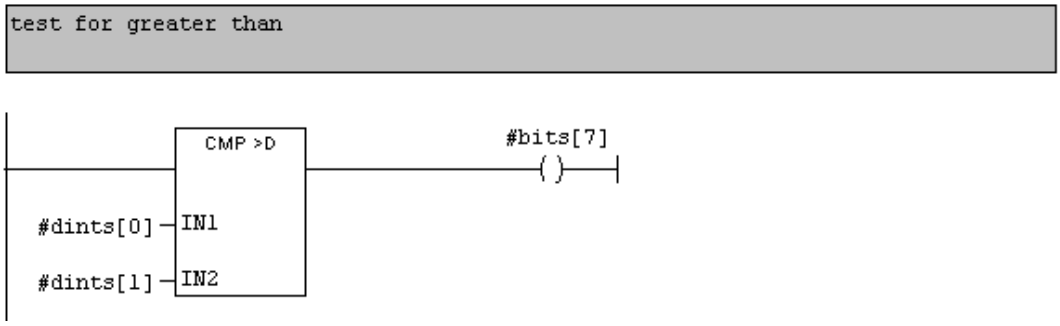
STEP 7



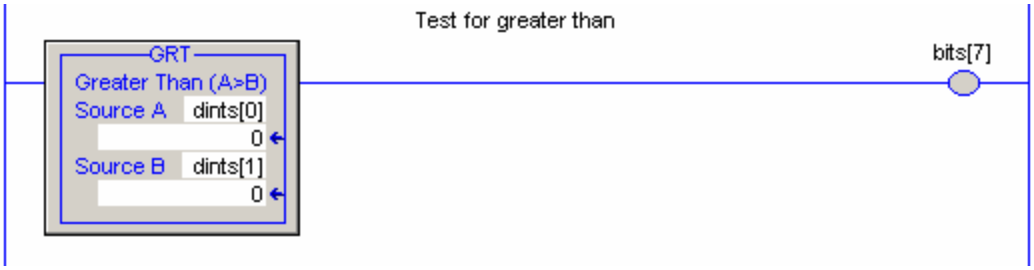
LOGIX

Test del valore maggiore di

STEP 7



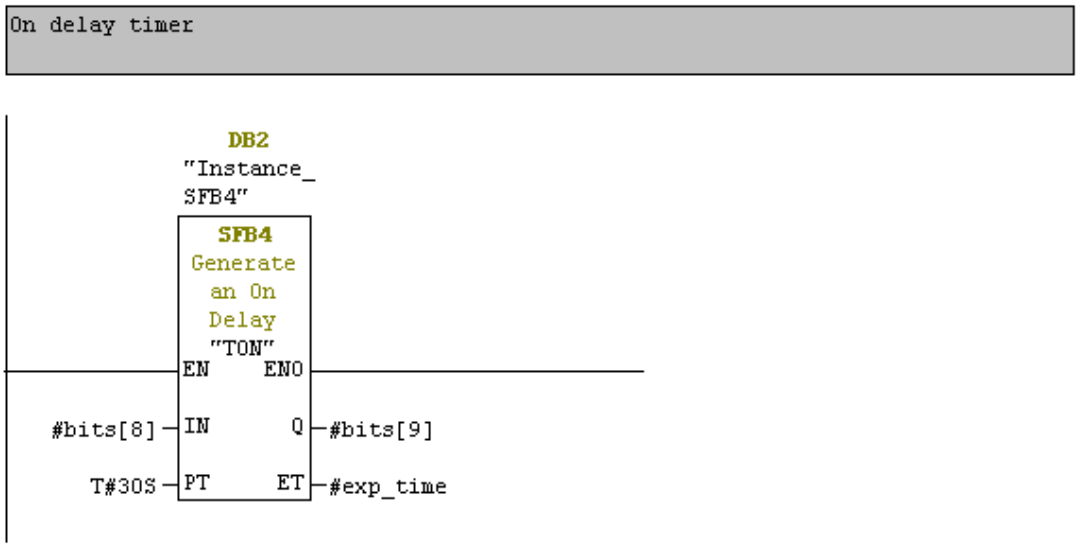
LOGIX



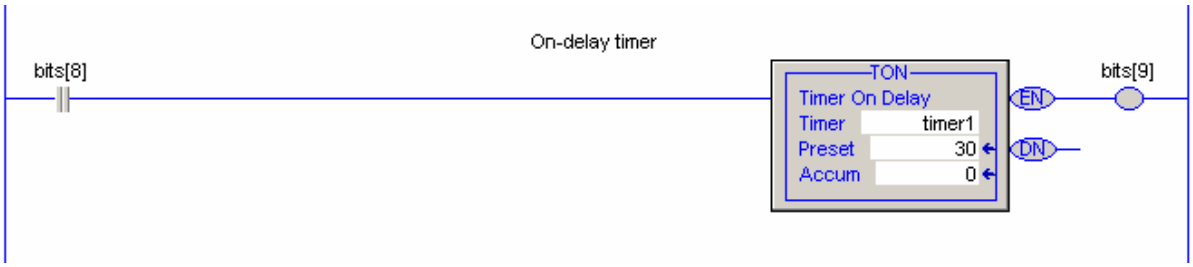
Come prima, utilizzare l'istruzione CMP se l'espressione è più complessa del semplice confronto tra due numeri.

Timer con ritardo all'eccitazione

STEP 7

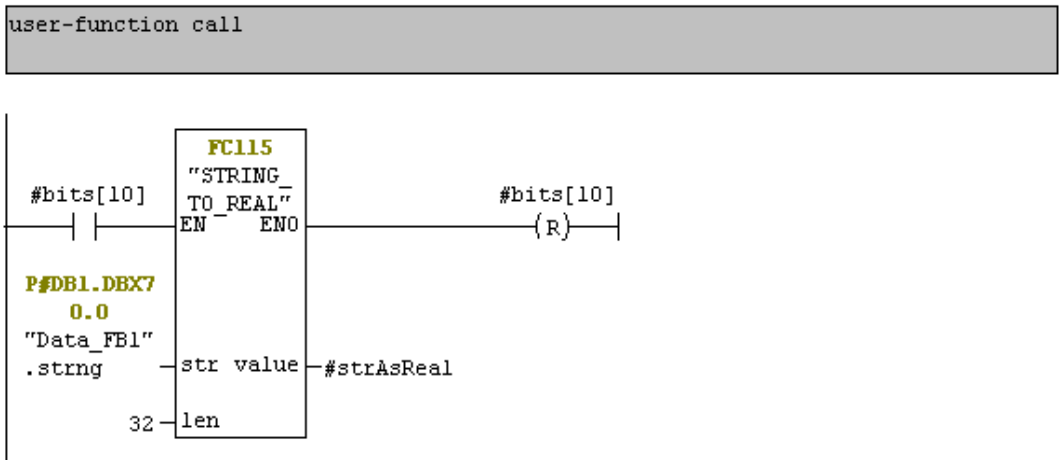


LOGIX

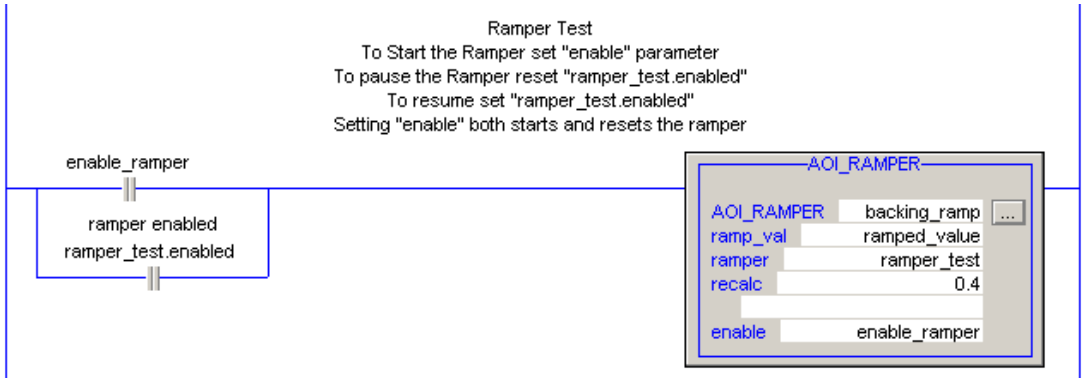


Chiamata alla funzione utente

STEP 7

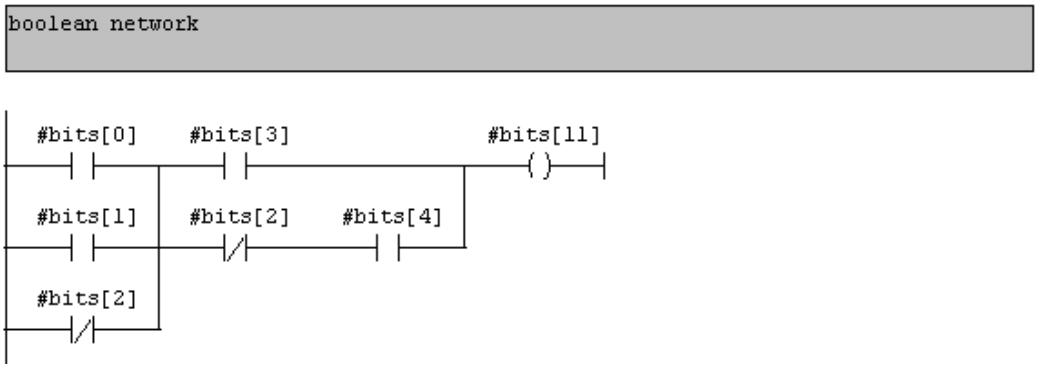


LOGIX

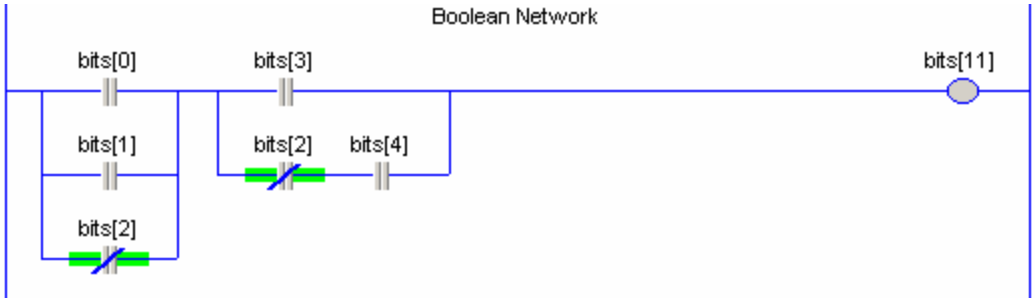


Rete booleana

STEP 7



LOGIX

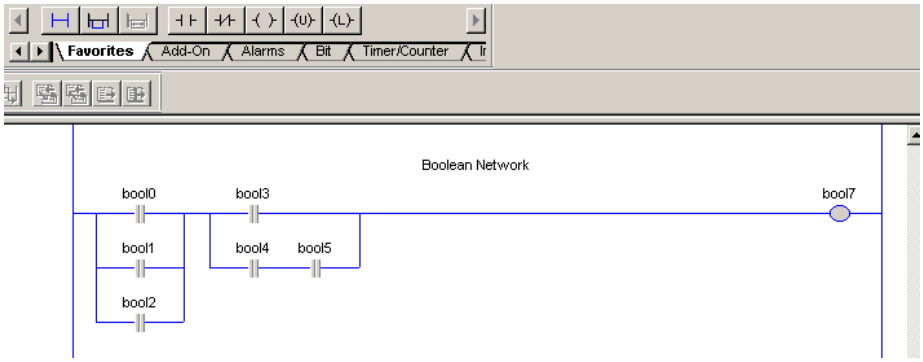


Le similitudini tra LAD di STEP 7 e LD di Logix sono sufficienti da rendere la traduzione a livello di routine abbastanza facile.

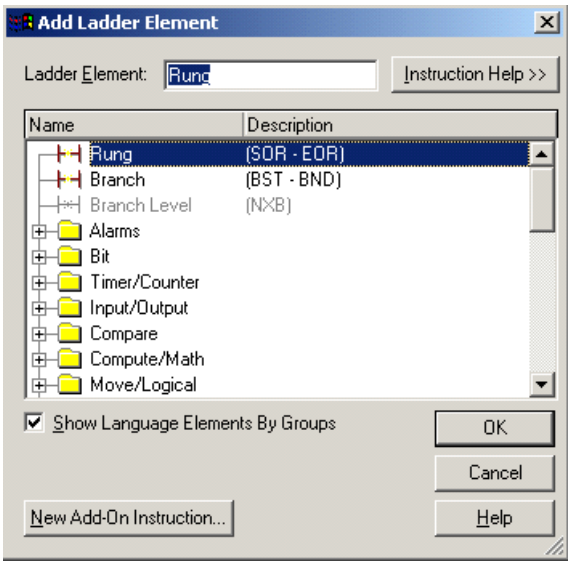
Editor LD di Logix

Sono disponibili almeno sette metodi per selezionare le istruzioni LD. Di seguito sono descritti due metodi abbastanza simili a quelli di STEP 7.

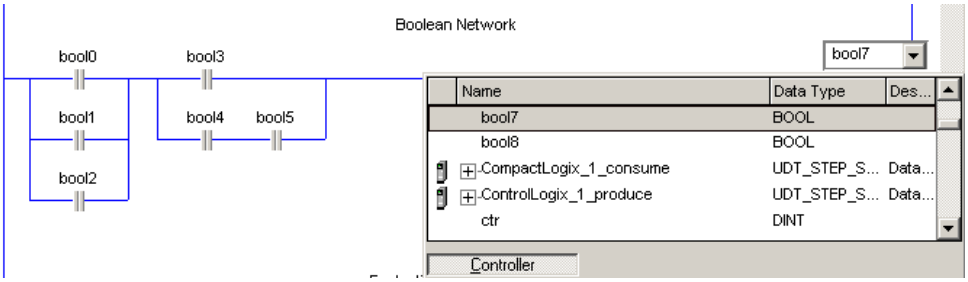
È possibile eseguire la selezione da una tavolozza sopra il foglio di lavoro LD.



Se si digita Alt+Ins, sarà visualizzata la finestra pop-up di selezione.



Quando si configurano le istruzioni, sono disponibili menu a discesa in cui è possibile selezionare il tag da immettere.



Salti e processo decisionale

STEP 7 - Sequenza di salti tradizionale

Il seguente task di esempio è spiegato nel commento di rete. Sono illustrate due versioni di S7 perché entrambe sono utilizzate spesso.

```
Network 1: Multi-way selection
if #input is 5 set #target to 8

else if #input is 6 set #target to 10

else if #input is 7 set #target to 16

else set #input to 0
```

```
      L     #input
      L     5
      ==I
      JCN   _001

      L     8
      T     #target

      JU    end

_001: L     #input
      L     6
      ==I
      JCN   _002

      L     10
      T     #target

      JU    end

_002: L     #input
      L     7
      ==I
      JCN   _003

      L     16
      T     #target

      JU    end

_003: L     0
      T     #target

end:  NOP   0
```

Il valore di #input viene confrontato con il set di costanti finché non viene soddisfatto il confronto. Dopodiché viene eseguita l'azione e il confronto termina. Se #input non combacia con nessun valore nel set, viene eseguita un'azione predefinita.

STEP 7 - Jump List

In questo esempio, il task è lo stesso, ma viene utilizzata un'istruzione Jump List, che è simile a una tabella di salti del microprocessore e trasferisce l'esecuzione a un'etichetta in base al valore di una variabile.

Network 2 : Title:

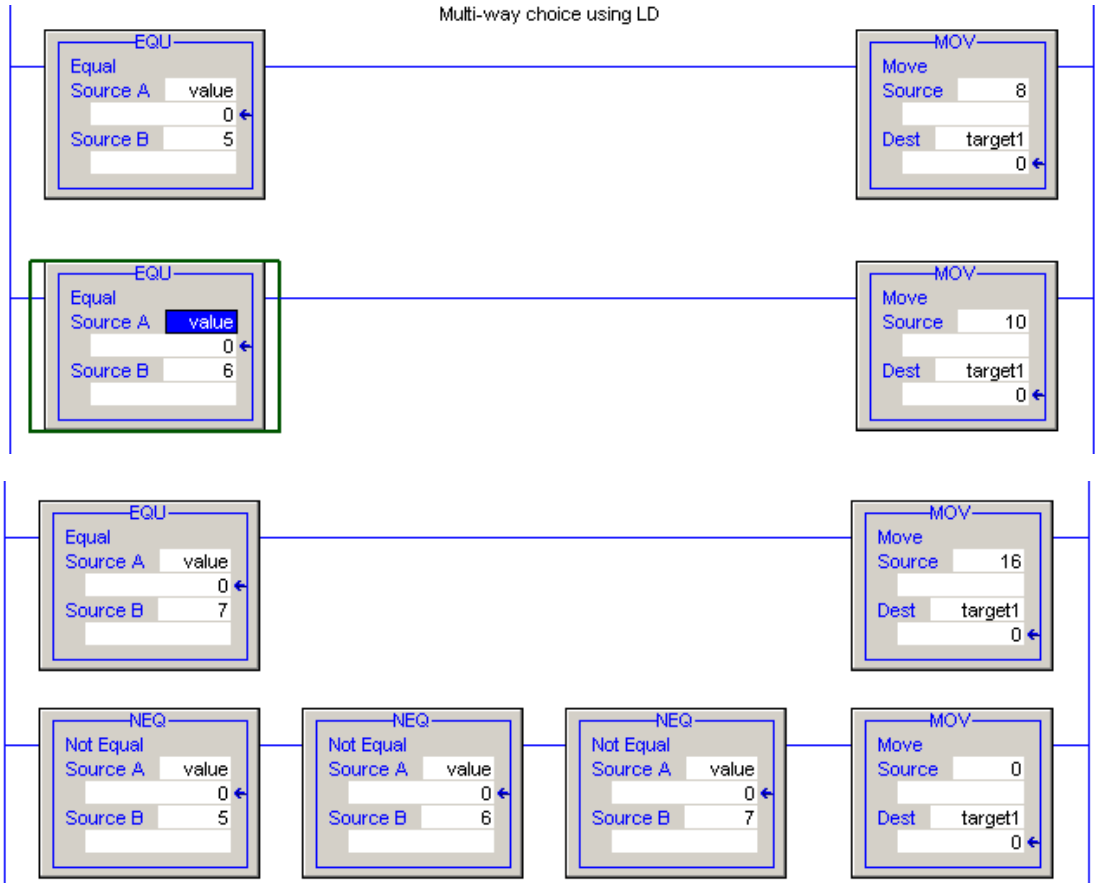
```
if #input is 5 set #target to 8

else if #input is 6 set #target to 10

else if #input is 7 set #target to 16

else set #input to 0
```

```
          L     #input
          L     5
          -I
          JL    rng
          JU    d5
          JU    d6
          JU    d7
    rng:  L     0
          T     #target
          JU    cont

    d5:   L     8
          T     #target
          JU    cont

    d6:   L     10
          T     #target
          JU    cont

    d7:   L     16
          T     #target

    cont: NOP   0
```

È più leggibile di una sequenza di salti tradizionale ed è più efficace perché viene eseguito solo il codice dall'etichetta di destinazione.

Logix - Logica ladder

È mostrata la scelta multipla utilizzando LD.



Logix - Testo strutturato If…Then…Else

Tutti gli utenti esperti di un linguaggio di programmazione delle famiglie Basic/Pascal/C comprenderanno questo esempio senza problemi.

```
// multi-way choice using Structured Text

if (value = 5) then target := 8;
elsif (value = 6) then target := 10;
elsif (value = 7) then target := 16;
else target := 0;
end_if;
```

Le parentesi che racchiudono la condizione “if” non sono obbligatorie.

Istruzione in testo strutturato CASE di Logix

Si tratta di un'altra variante in ST che esegue lo stesso task. Sufficientemente compatta e semplice non richiede molti commenti aggiuntivi.

```
// multi-way choice using Structured Text CASE

case value of
    5: target := 8;
    6: target := 10;
    7: target := 16;
else target :=0;
end_case;
```

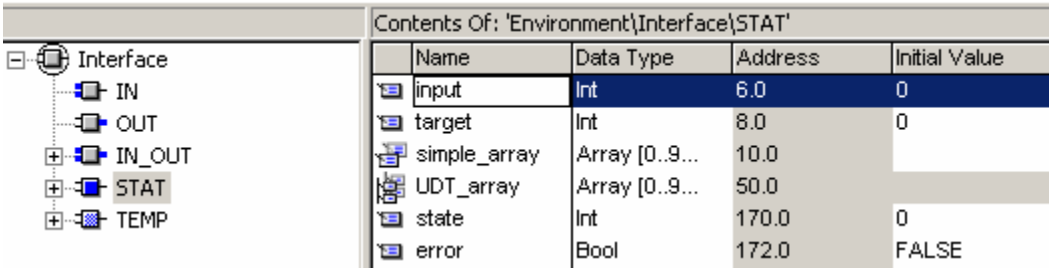
Tutte le soluzioni funzionano, tuttavia questa è la soluzione Logix preferenziale. È compatta e sufficientemente semplice da non richiedere ulteriore documentazione.

Array

Sia STEP 7 che Logix consentono la creazione in memoria di array di oggetti semplici o complessi. Logix dispone di un supporto di alto livello per l'accesso agli array. In STEP 7, tuttavia, è necessaria la programmazione di basso livello.

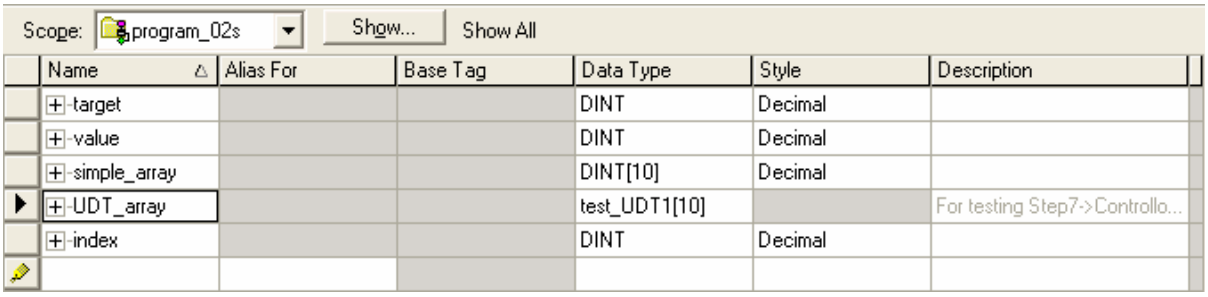
Creazione di array in STEP 7

Nella seguente immagine sono mostrati due array creati in un blocco dati dell'istanza. Simple_array è un array di 10 elementi. UDT_array è un array di 10 strutture del tipo test_UDT1, dove test_UDT1 è un tipo di dati utente contenente alcuni altri tipi non mostrati.



Creazione di array in Logix

La procedura è esattamente la stessa in Logix.



Sintassi di dichiarazione array

STEP 7 utilizza la sintassi di dichiarazione ARRAY[0...15] OF REAL. Logix utilizza REAL[15].

STEP 7 ha una sintassi speciale per le stringhe. STRING[32] è una stringa a 32 caratteri in STEP 7, mentre STRING[32] di Logix è un array di trentadue stringhe, ciascuna contenente 82 caratteri.

Accesso agli array in STEP 7

Il seguente esempio è relativo all'esecuzione di un task semplice su due array simple_array[] e UDT_array[]. Il task è descritto nel commento di rete.

In STEP 7, non è possibile accedere agli array utilizzando l'annotazione standard array[]. È invece necessario utilizzare le operazioni di basso livello con i puntatori. Nel frammento riportato di seguito, una funzione "GET_INDEXED_REFERENCE" facilita notevolmente il task restituendo un puntatore all'elemento dell'array al quale accedere.

```
Network 3 : Title:
array operations
----------------
if (simple_array[2] = simple_array[5]) then
  UDT_array[8].booleanl := 1;
else
  UDT_array[8].booleanl := 0;
end_if;

    // 1.  compare simple_array[2] with simple_array[5]
          CALL  "GET_INDEXED_REFERENCE"           FC111
           refArray  :="Data_test".simple_array   P#DB1.DBX10.0
           index     :=2
           byteIncr  :=4
           startIndex:=FALSE
           retVal    :=#ptr1                          BOOL
          CALL  "GET_INDEXED_REFERENCE"           FC111
           refArray  :="Data_test".simple_array   P#DB1.DBX10.0
           index     :=5
           byteIncr  :=4
           startIndex:=FALSE
           retVal    :=#ptr2
          OPN   "Data_test"                       DB1
          L     DID [#ptr1]
          L     DID [#ptr2]
          ==D
          =     #compare

   // 2. get pointer to UDT_array
          CALL  "GET_INDEXED_REFERENCE"           FC111
           refArray  :="Data_test".UDT_array      P#DB1.DBX44.0
           index     :=8
           byteIncr  :=12    INT
           startIndex:=FALSE
           retVal    :=#ptr1
          L     #ptr1
          LAR1

   // 3. set or reset the bit
          A     #compare
          =     DIX [AR1,P#0.0]
```

In tal caso, il codice effettivo del testo strutturato di Logix è stato utilizzato come commento di rete, dimostrando così l'elevata immediatezza del codice Logix.

STEP 7 - Loop tra elementi degli array

L'obiettivo di questo esempio è eliminare il campo a virgola mobile in ciascuna struttura in UDT_array[]. Questa operazione è di facile esecuzione, tuttavia è richiesta esperienza nell'utilizzo dei puntatori.

```
Array Operations
----------------
Clear all float elements at UDT offset P#6.0 in array UDT_array
```

```
// transfer pointer to UDT_array to AR1
      L     P##UDT_array
      LAR1
// initialise counter
      L     0
      T     #ctr

// end if #ctr > 9
loop: L     #ctr
      L     9
      >I
      JC    end2
// clear the float field at offset p#6.0
      L     0.000000e+000
      T     DID [AR1,P#6.0]
// increment AR1 by size of the UDT
      +AR1  P#12.0
// increment counter
      L     #ctr
      INC   1
      T     #ctr
// loop back
      JU    loop

end2: NOP   0
```

Logix - Operazioni degli array nel testo strutturato

Nel seguente frammento di ST vengono eseguiti i task descritti nelle due sezioni precedenti.

```
// array access in ST
if (simple_array[2] = simple_array[5]) then
    UDT_array[8].booleanl := 1;
else
    UDT_array[8].booleanl := 0;
end_if;


// clearing array elements
if (simple_array[0] = 5) then
    index := 0;
    while (index <= 9) do
        UDT_array[index].reall := 0.0;
        index := index + 1;
    end_while;
end_if;
```

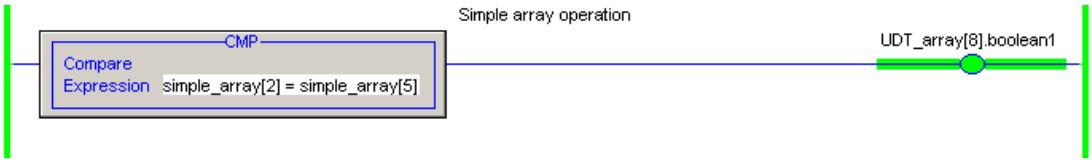
Non sono necessari commenti aggiuntivi per descriverne la modalità di funzionamento.

Se è necessario gestire le variabili booleane con le istruzioni if...then...else, prendere in considerazione la scrittura di un'equazione booleana:

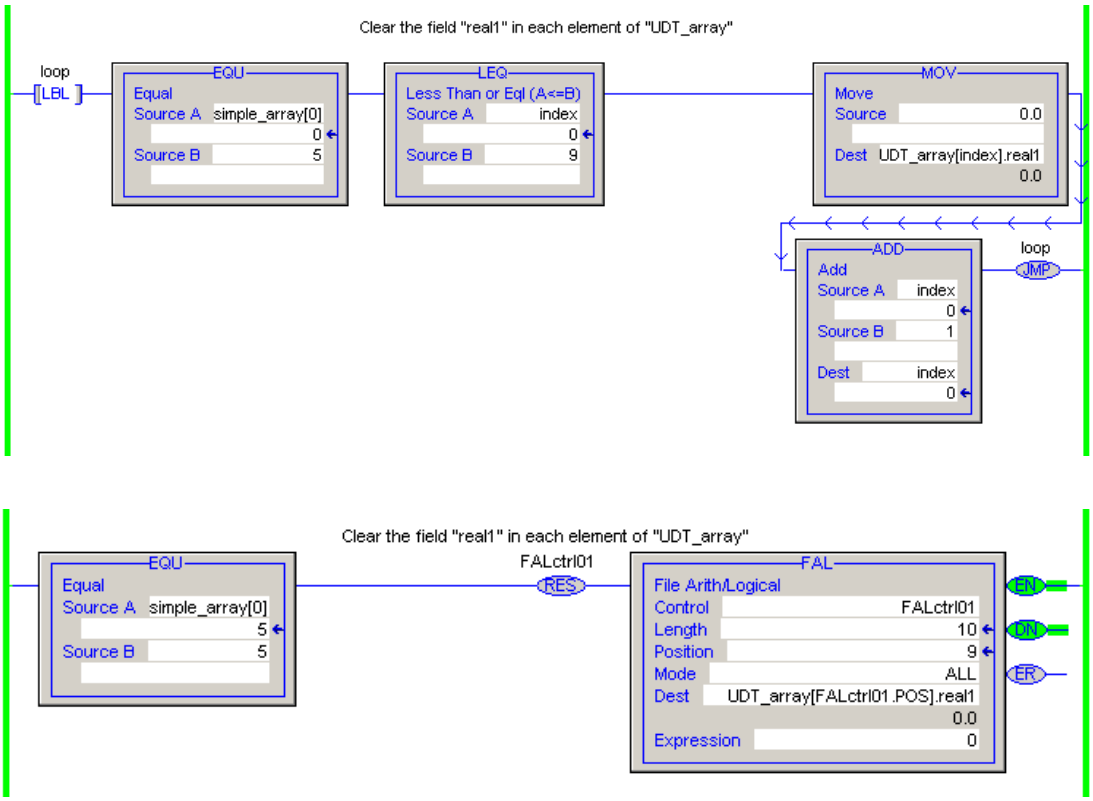
```
// array access in ST
UDT_array[8].booleanl := simple_array[2] = simple_array[5];
```

Logix - Operazioni di array nel linguaggio ladder

Gli esempi della sezione precedente possono essere scritti in LD utilizzando l'istruzione CMP (Compare) come segue.



Il secondo esempio (azzeramento del campo ~~reale~~ nell'array degli UDT) può essere eseguito in uno dei seguenti modi.

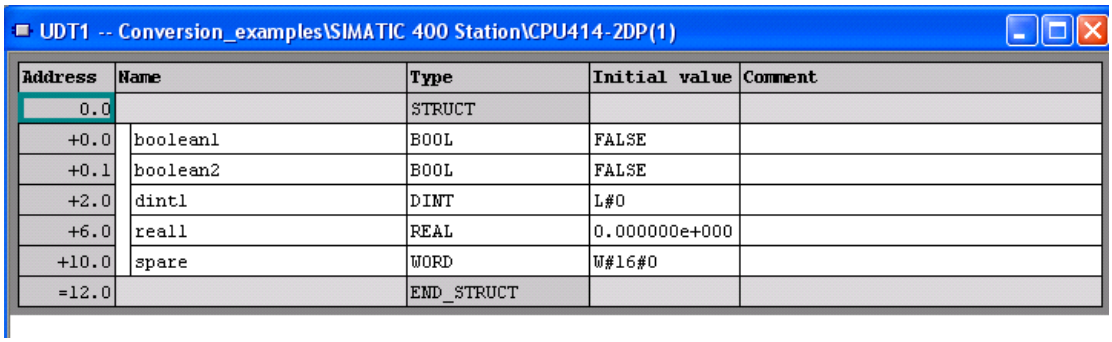


Il primo approccio all'azzeramento degli elementi dell'array è una traduzione dal loop while del codice ST. Il secondo approccio impiega l'istruzione FAL avanzata per le operazioni degli array.

Tipi di dati utente

Configurazione ed utilizzo dei tipi di dati utente (UDT, User Data Type) in STEP 7 e Logix sono molto simili.

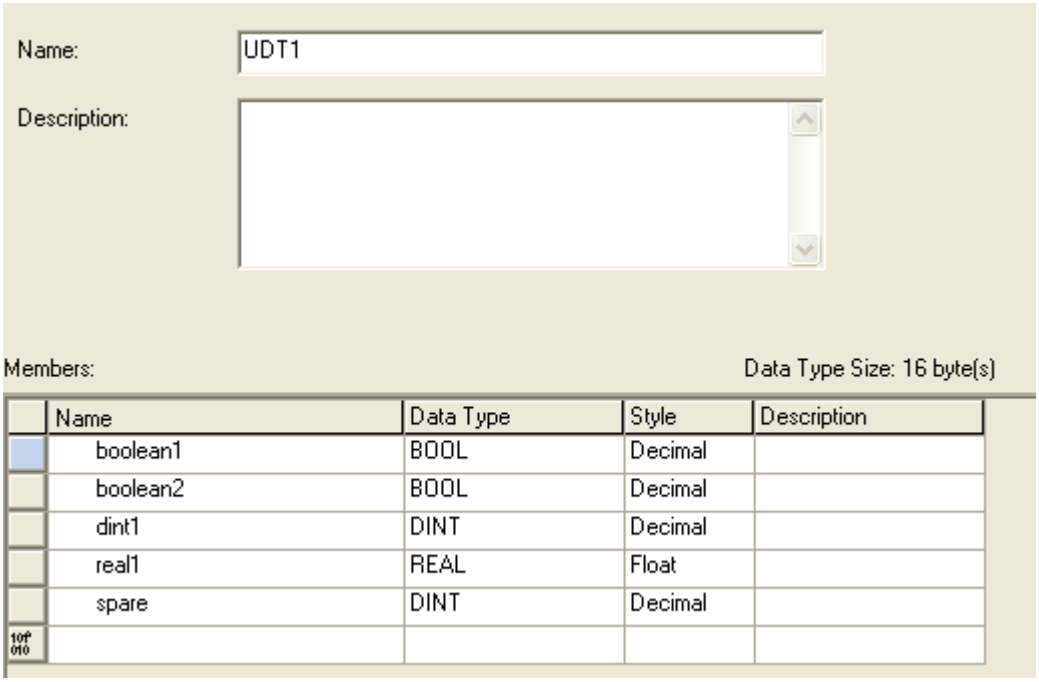
Di seguito è riportato un UDT in STEP 7.



UDT1 -- Conversion_examples\SIMATIC 400 Station\CPU414-2DP(1)

Address	Name	Type	Initial value	Comment
0.0		STRUCT		
+0.0	boolean1	BOOL	FALSE	
+0.1	boolean2	BOOL	FALSE	
+2.0	dint1	DINT	L#0	
+6.0	real1	REAL	0.000000e+000	
+10.0	spare	WORD	W#16#0	
=12.0		END_STRUCT		

Di seguito è riportato un UDT in Logix.



Name: UDT1

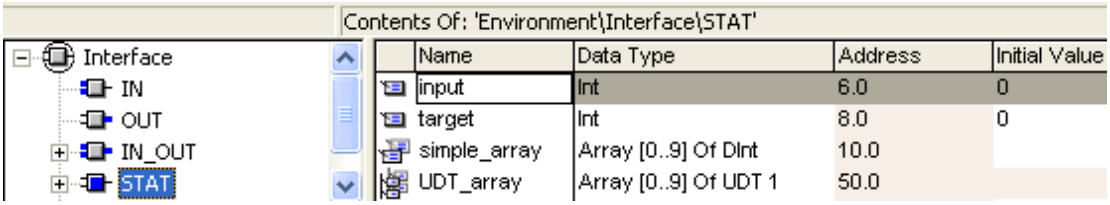
Description:

Members: Data Type Size: 16 byte(s)

	Name	Data Type	Style	Description
	boolean1	BOOL	Decimal	
	boolean2	BOOL	Decimal	
	dint1	DINT	Decimal	
	real1	REAL	Float	
	spare	DINT	Decimal	

In entrambi i sistemi, gli UDT possono essere utilizzati per dichiarare e definire le variabili.

Di seguito è riportata una dichiarazione relativa a un UDT in STEP 7.

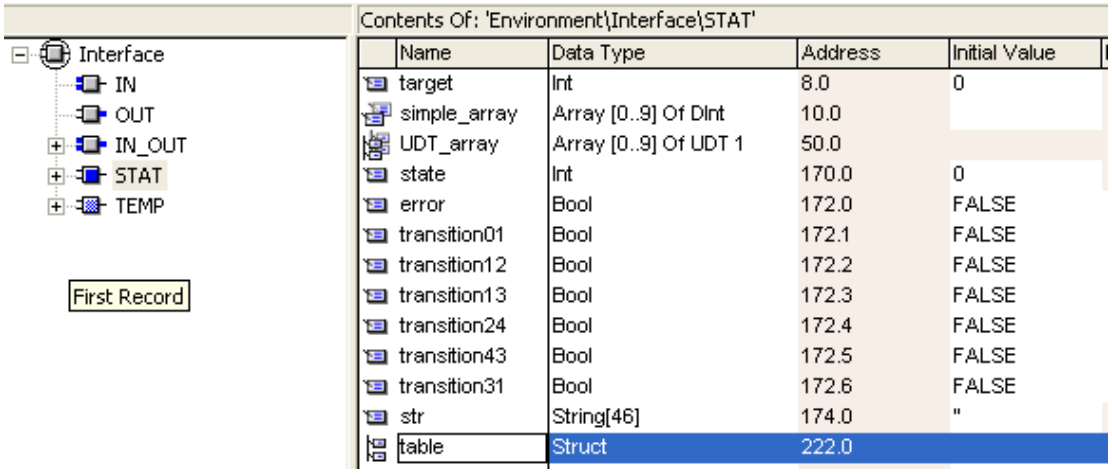


Di seguito è riportata una dichiarazione relativa a un UDT in Logix.

Name	Alias For	Base Tag	Data Type	Style
Limit_Switch_1	Local:3:I.Data.0	Local:3:I.Data.0	BOOL	Decimal
Local:3:C			AB:1756_DI:C:0	
Local:3:I			AB:1756_DI:I:0	
conveyor_1			UDT1	
conveyor_1.boolean1			BOOL	Decimal
conveyor_1.boolean2			BOOL	Decimal
conveyor_1.dint1			DINT	Decimal
conveyor_1.real1			REAL	Float
conveyor_1.spare			DINT	Decimal

La seguente è una differenza secondaria tra i due sistemi:

In STEP 7 è possibile dichiarare una variabile del tipo “struct”.



Si noti la voce “table” del tipo Struct. All'interno di “table” può essere presente una raccolta (ordinata o non ordinata) di qualsiasi combinazione di tipi.

In Logix, questa operazione verrebbe eseguita definendo “Struct” come un UDT contenente la struttura dati richiesta, quindi dichiarando “table” come tipo Struct.

Name	Alias For	Base Tag	Data Type	Style
Limit_Switch_1	Local:3:I.Data.0	Local:3:I.Data.0	BOOL	Decimal
⊞ Local:3:C			AB:1756_DI:C:0	
⊞ Local:3:I			AB:1756_DI:I:0	
⊟ conveyor_1			UDT1	
conveyor_1.boolean1			BOOL	Decimal
conveyor_1.boolean2			BOOL	Decimal
⊞ conveyor_1.dint1			DINT	Decimal
conveyor_1.real1			REAL	Float
⊞ conveyor_1.spare			DINT	Decimal
⊟ table			Struct	
table.status_bit1			BOOL	Decimal
table.status_bit2			BOOL	Decimal
table.status_bit3			BOOL	Decimal
⊞ table.dwell_timer1			TIMER	
⊞ table.dwell_timer2			TIMER	
table.speed			REAL	Float

Puntatori e array

Un programma STEP 7 può avere puntatori a qualsiasi oggetto di dati. È inoltre consentito l'accesso indiretto ai blocchi dati, tuttavia non sono presenti puntatori alle funzioni (ad eccezione di una modalità limitata dall'istruzione JL (Jump List)). Il puntatore dati è insolito, poiché è un puntatore a un bit. Pertanto il relativo valore è otto volte maggiore rispetto a quello di un normale puntatore a un byte. Ciò riflette l'importanza dei bit nella programmazione dei sistemi di controllo.

In Logix non esistono puntatori. Gli array eseguono la stessa funzione dei puntatori, ma sono più semplici e sicuri.

Un programmatore di S7 sarà in grado di eseguire un'intera gamma di task in Logix senza puntatori? Nella programmazione dei computer, i puntatori ai dati sono utilizzati principalmente per tre scopi:

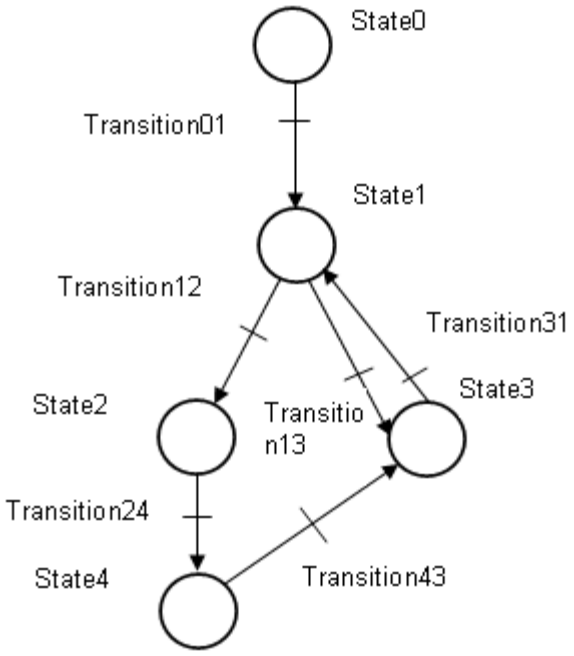
- operazioni sugli elementi di dati ordinati in modo sequenziale (array di oggetti, stringhe)
- assegnazione, accesso ed eliminazione degli oggetti assegnati in modo dinamico
- passaggio dei riferimenti agli oggetti come parametri nelle chiamate di funzione.

In Logix, il primo scopo è coperto dagli array. Il secondo scopo non è attinente al software di controllo poiché non esistono oggetti assegnati in modo dinamico. Il terzo scopo è coperto dai parametri “inout” nei blocchi funzione di STEP 7 e nelle istruzioni add-on di Logix.

Se ne conclude pertanto che l'assenza di puntatori espliciti non è una limitazione per i programmatori di Logix. I programmatori di STEP 7 potrebbero inoltre comprendere che la codifica con gli array può essere eseguita più velocemente nel testo strutturato che utilizza array rispetto a STL che utilizza puntatori.

Macchina a stati

La macchina a stati è un costrutto importante del software dei sistemi di controllo poiché semplifica notevolmente il task di programmazione del controllo sequenziale.



Macchina a stati di STEP 7

STEP 7 offre un diagramma funzionale sequenziale grafico come aggiunta opzionale all'applicazione di base. Se l'SFC grafico non è disponibile, l'operazione verrà eseguita dalla lista istruzioni.

```
Network 4: Title:
state machine

      L     #state
      JL    rng1
      JU    st0
      JU    st1
      JU    st2
rng1: SET
      S     #error
      BEU

st0:  L     1
      A     #transition01
      JC    next
      JU    ovr

st1:  L     2
      A     #transition12
      JC    next

      L     3
      A     #transition13
      JC    next
      JU    ovr

st2:  L     4
      A     #transition24
      JC    next
      JU    ovr

st3:  L     0
      A     #transition31
      JC    next
      JU    ovr

st4:  L     3
      A     #transition43
      JC    next
      JU    ovr

next: T     #state

ovr:  NOP   0
```

La variabile #state contiene il numero di stato. L'istruzione Jump List forza l'esecuzione a saltare all'etichetta relativa al valore di #state. Se una condizione di transizione da tale stato è True, il nuovo valore di stato viene caricato nell'accumulatore e l'esecuzione salta all'etichetta “next”, dove il nuovo numero di stato viene trasferito alla variabile #state.

Macchina a stati di Logix nel testo strutturato

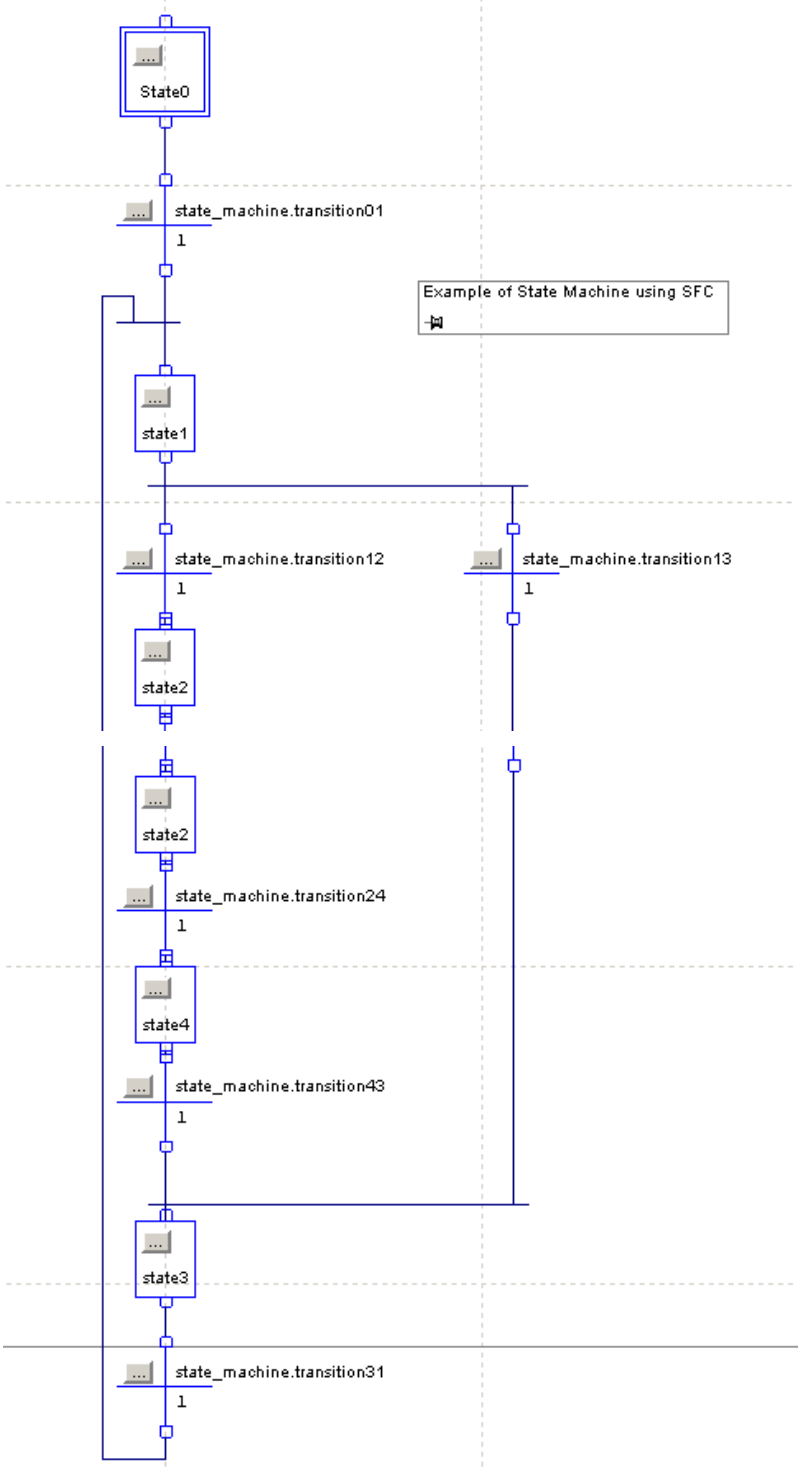
In questo esempio è riportata la stessa macchina a stati nel testo strutturato, che utilizza l'istruzione CASE. Come per altri esempi di ST, risulterebbe difficile scrivere una descrizione più chiara del codice stesso.

```
// implementation of State Machine using CASE in ST

case state_machine.state of
    0:  if state_machine.transition01 then
            state_machine.state := 1;
        end_if;
    1:  if state_machine.transition12 then
            state_machine.state := 2;
        elsif state_machine.transition13 then
            state_machine.state := 3;
        end_if;
    2:  if state_machine.transition24 then
            state_machine.state := 4;
        end_if;
    3:  if state_machine.transition31 then
            state_machine.state := 1;
        end_if;
    4:  if state_machine.transition43 then
            state_machine.state := 3;
        end_if;
end_case;
```

Macchina a stati di Logix nel diagramma funzionale sequenziale

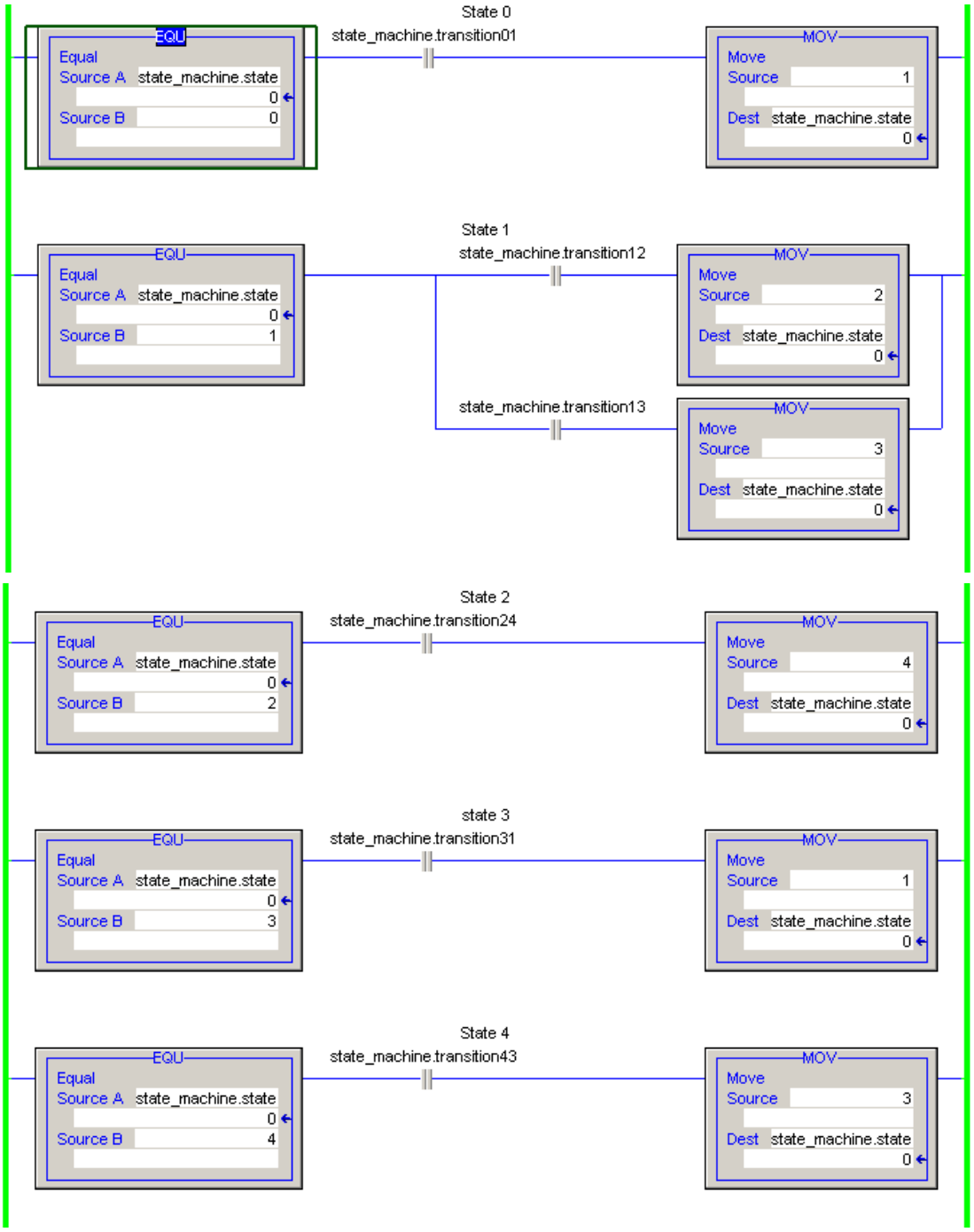
Logix fornisce un SFC grafico come una delle sue suite di linguaggi standard. Di seguito è mostrata una macchina a stati in SFC.

Implementazione della macchina a stati utilizzando il grafico SFC



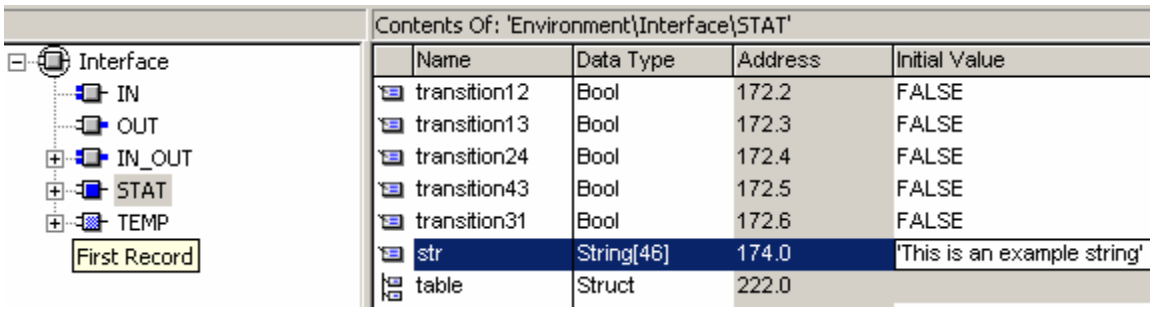
Macchina a stati nel linguaggio ladder

Nell'immagine seguente è mostrato in che modo la macchina a stati può essere implementata in LD.



Stringhe

Definizione di stringhe in STEP 7



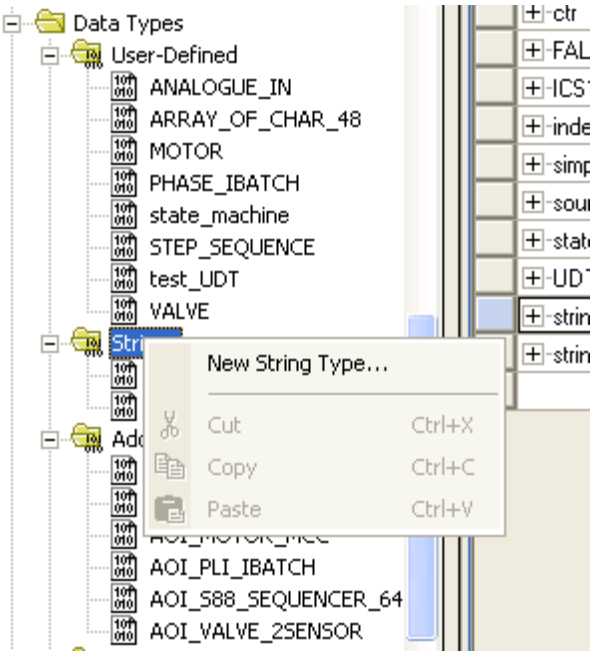
L'intestazione dei dati indica in che modo vengono definite le stringhe. La lunghezza della stringa viene immessa tra parentesi [] dopo il tipo di dati String. Il valore iniziale della stringa è digitato nella colonna "Initial Value".

È possibile creare un array di stringhe, tuttavia non è possibile assegnare a ciascuna un valore iniziale. Una definizione alternativa per evitare questo problema è mostrata dalla voce "table" nell'intestazione dati. "Table" è una struttura. I contenuti della struttura, non mostrati, sono cinque istanze di string[46], a ciascuna delle quali è stato assegnato un valore iniziale.

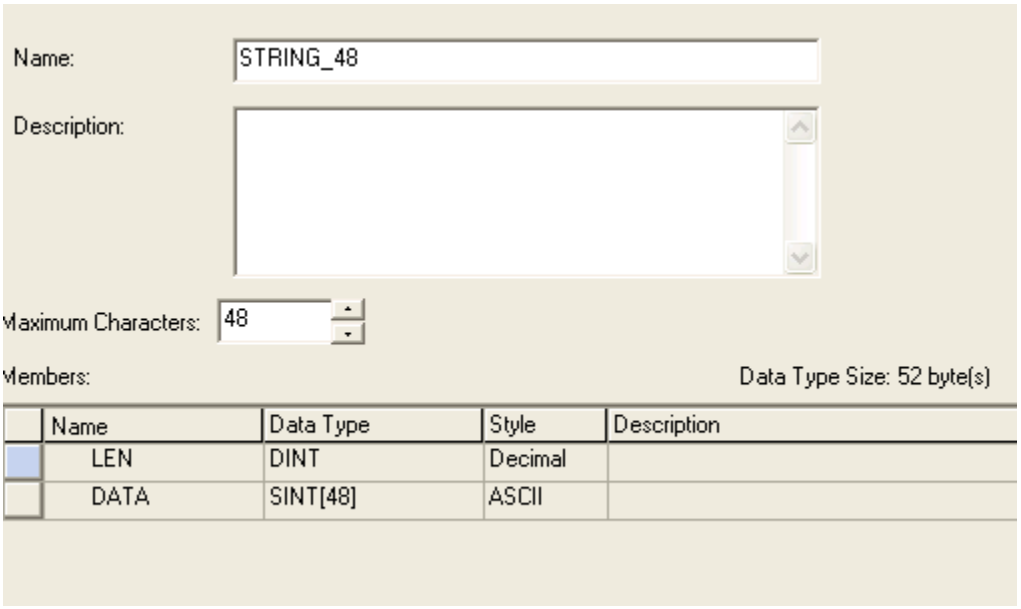
Definizione di stringa in Logix

L'estratto dalla tabella di configurazione dei tag di seguito mostra in che modo le stringhe sono definite in Logix.

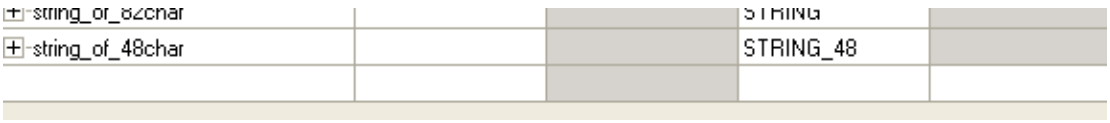
Se si desidera creare una stringa di lunghezza diversa da quella predefinita di 82 caratteri, fare clic con il pulsante destro del mouse su “strings” nell'albero del progetto (come mostrato di seguito).



Configurare quindi le proprietà come segue.



Dopo avere eseguito questa operazione, è possibile definire le istanze del nuovo tipo.



Con le istanze del tipo STRING o STRING_48, è presente un campo LEN che si aggiorna automaticamente quando viene immessa una costante della stringa o quando la stringa è manipolata dalle istruzioni ASCII o STRING.

Variabili temporanee di STEP 7

Una delle categorie di variabili in STEP 7 è la variabile temporanea, che può essere creata in qualsiasi blocco organizzativo, funzione o blocco funzione.

Le variabili temporanee sono utilizzate per l'archiviazione temporanea locale dei valori intermedi e per i puntatori. Esistono solo quando il relativo blocco è in esecuzione, e i valori associati vengono persi al termine del blocco.

Logix non dispone di variabili temporanee. Tutta l'archiviazione è statica, ovvero i valori sono conservati tra le esecuzioni del codice.

Se si utilizzano istruzioni add-on, si noterà che è possibile creare variabili locali per un'istruzione add-on. Queste variabili possono essere utilizzate allo stesso modo delle variabili temporanee.

Funzioni

Se il programmatore di STEP 7 utilizza la lista istruzioni, potrebbe dover sviluppare routine di basso livello, che richiedono molto tempo per la scrittura e richiedono test minuziosi. Le funzioni sono importanti perché lo sviluppo di tali routine deve essere eseguito solo una volta, dopodiché sia il creatore della funzione che gli altri programmatori possono eseguire le stessa operazione in una frazione del tempo.

In questa sezione viene illustrato in che modo è possibile implementare le funzioni in Logix.

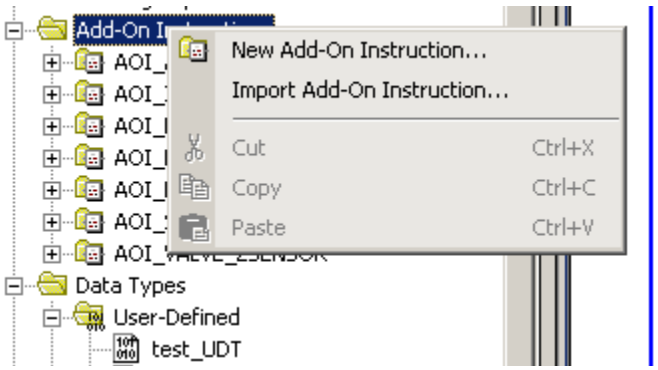
Funzione come istruzione add-on in Logix

Le funzioni e i blocchi funzione in STEP 7 hanno una struttura simile all'istruzione add-on in Logix. L'istruzione add-on dispone degli stessi tipi di parametri di FB (Input, Output e InOut) e della propria area dati. Una volta codificata e testata, l'istruzione add-on può essere utilizzata da qualsiasi parte del programma ed è sufficientemente autonoma da essere esportata in altri progetti o inserita in una libreria del codice.

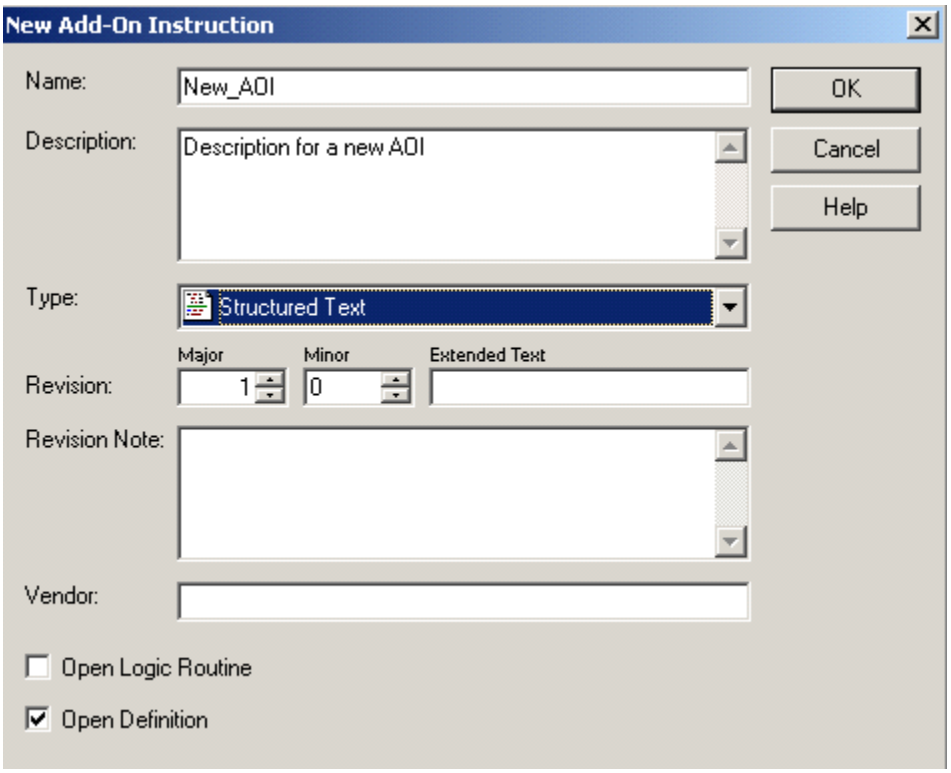
Esempio - Funzione di rampa

In questo esempio, una variabile reale viene aumentata a rampa in modo lineare dal valore corrente al nuovo valore alla velocità specificata.

Accedere al ramo delle istruzioni add-on dell'albero del progetto e fare clic con il pulsante destro del mouse sull'istruzione add-on.

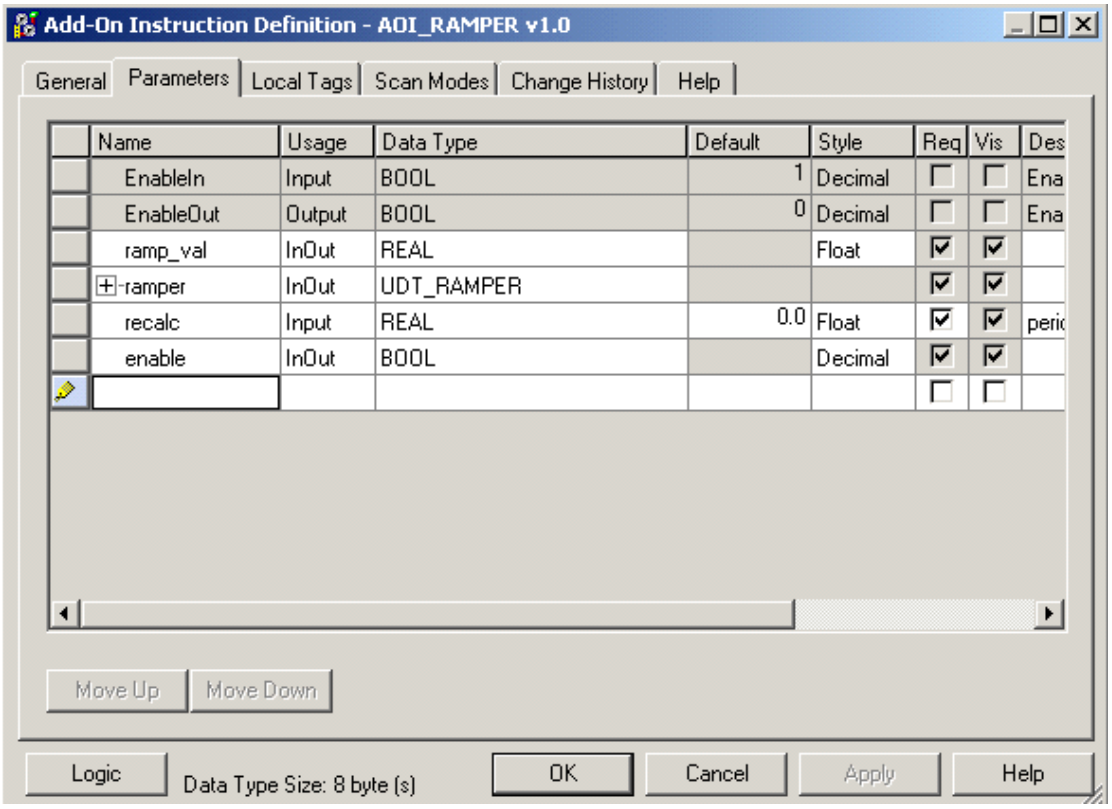


Viene visualizzato il seguente modulo.

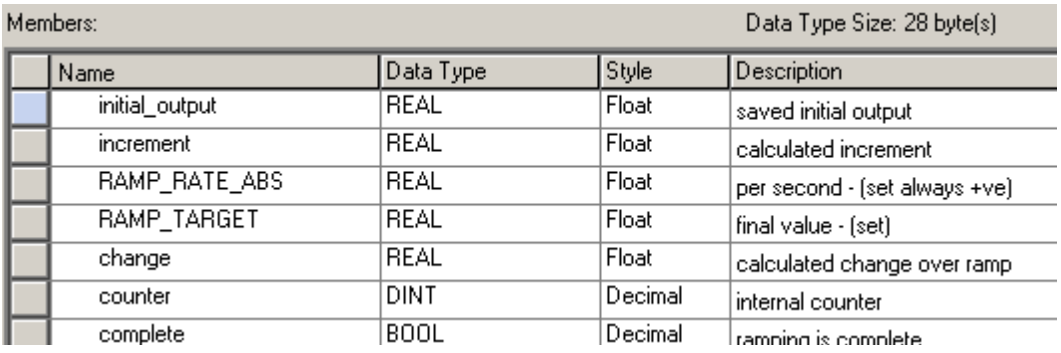


Immettere il nome dell'istruzione add-on e specificare il linguaggio in cui sarà scritta la relativa sezione del codice.

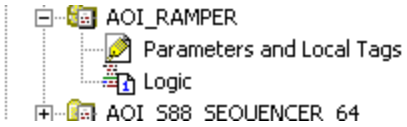
Scegliere la scheda Parameters.



Come in STEP 7, i parametri Input sono valori dal programma all'istruzione add-on, i parametri Output sono valori dall'istruzione add-on al programma e i parametri InOut sono per le variabili che saranno modificate dell'istruzione add-on. Se si dispone di una qualsiasi struttura di dati, scegliere comunque il tipo InOut poiché i parametri sono passati per riferimento ed è la soluzione più efficace.



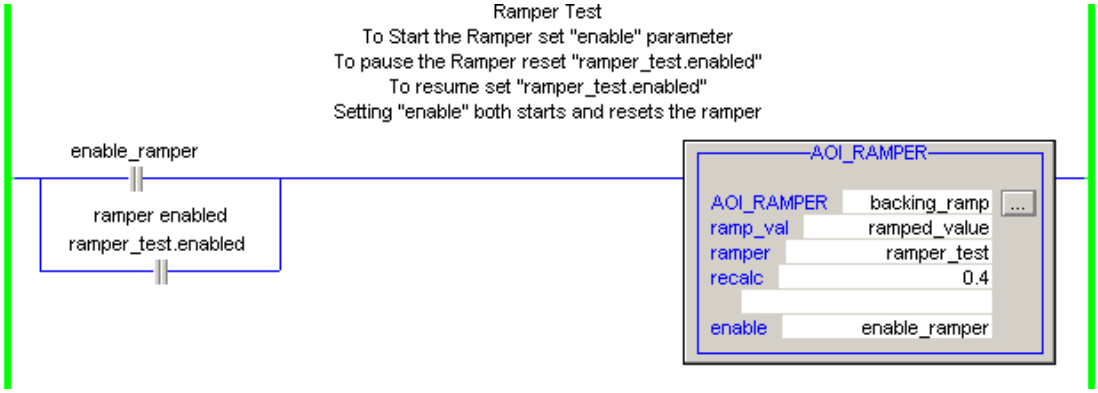
In questa directory di progetto per AOI_RAMPER, è presente una sezione logica.



Aprirla per visualizzare il codice relativo a questa istruzione add-on.

```
// Ramps a real variable from its current valuse to a new value at a
// specified rate.

// Parameters:
//      ramp_val    - variable to be ramped
//      ramper      - instance of UDT UDT_RAMPER
//      recalc      - code recalculation period (s)
//      enable      - start signal

// To use - set the target value in ramper.RAMP_TARGET_ABS
//        - set the ramp rate in ramper.RAMP_RATE_ABS
//        - to Start the Ramper set "enable" parameter
//        - to pause the Ramper reset "ramper.enabled"
//        - to resume set "ramper.enabled"
//        - setting "enable" both starts and resets the ramper

// on completion, the UDT field "complete" is set and the UDT field
// "enabled" is reset

// when enable is set, initialise
if (enable & (enable xor ramper._enable)) then
    ramper.initial_output := ramp_val;
    ramper.change := ramper.RAMP_TARGET - ramp_val;
    ramper.increment := ramper.change / abs(ramper.change)
                              * ramper.RAMP_RATE_ABS * recalc;
    ramper.counter := 0;
    ramper.complete := 0;
    enable := 0;
    ramper.enabled := 1;
end_if;

ramper._enable := enable;

// ramp calculations
if (ramper.enabled) then
    ramp_val := ramper.initial_output + (ramper.counter
                                            * ramper.increment);
    ramper.counter := ramper.counter + 1;
    if  (abs(ramper.counter * ramper.increment)
                                          > abs(ramper.change)) then
         ramp_val := ramper.RAMP_TARGET;
         ramper.complete := 1;
         ramper.enabled := 0;
    end_if;
end_if;
```

L'istruzione add-on può essere chiamata da qualsiasi routine.



Si noti che con le istruzioni add-on, sarà necessario creare un tag del tipo di istruzione add-on in un'area dati visibile nella routine, chiamato tag di supporto.

Prima di scrivere un'istruzione add-on, consultare la Guida alle istruzioni nel software RSLogix 5000. È possibile che esista già un'istruzione che eseguirà l'operazione. Nella seguente sezione viene illustrato questo argomento.

Copia di blocco, COP e CPS

In STEP 7, è comune utilizzare la funzione di sistema SFC20 “BLKMOV” per copiare un blocco di dati tra una posizione e l'altra.

```
CALL  "BLKMOV"
 SRCBLK :="Data_EM1".stepMsgs.step5
 RET_VAL:=#intVar
 DSTBLK :="Data_EM1".actualStep
```

L'istruzione copia la stringa dalla quinta posizione in un array di stringhe in una stringa di destinazione.

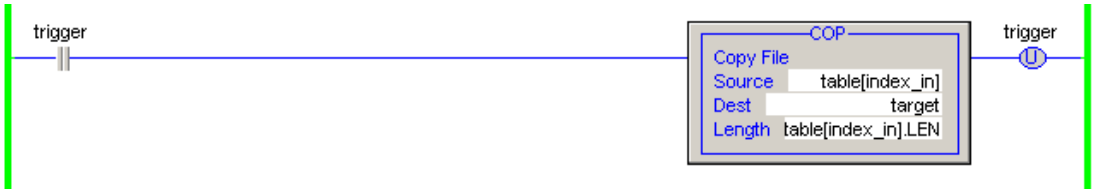
Spesso è necessario copiare l'elemento i da un array, dove “i” può variare durante l'esecuzione del programma. “BLKMOV” non è in grado di eseguire questa operazione.

Il programmatore di STEP 7 scrive una funzione per soddisfare il proprio requisito.

```
// copy step number descriptor to SCADA display area (EM faceplate)
      CALL  "INDEXED_COPY"
       indexSource :="Data_EMs".EM1.stepNumber
       sourceRef   :="Data_EM1".stepMsgs
       indexDest   :=1
       destRef     :="Data_EM1".actualStep
       recordLength:=8                  // 32 bytes
```

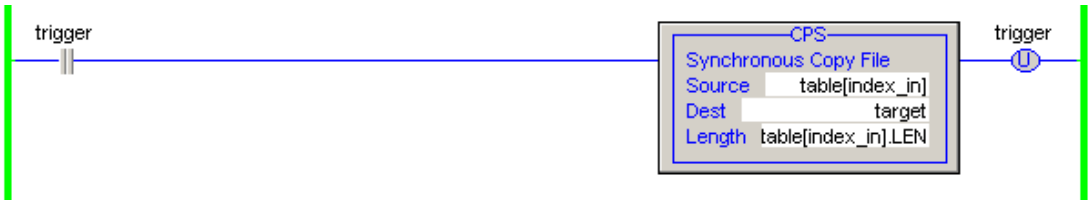
In questo caso, la copia viene eseguita tra due array e gli indici sono definiti da indexSource e indexDest.

In Logix, l'istruzione incorporata COP salverà tutto il lavoro.



Poiché le specifiche di origine e destinazione possono includere indici di array variabili, COP eseguirà il lavoro. È l'equivalente di “INDEXED_COPY”.

L'istruzione CPS è la stessa di COP, con un'unica differenza.



L'istruzione non può essere interrotta. Pertanto, i dati di origine e destinazione rimarranno costanti durante tutta l'esecuzione. Se si desidera spostare dati che potrebbero variare, utilizzare CPS.

Esempi:

- copia dei dati di ingresso in un buffer, dal quale il programma agirà sui dati.
- copia dei tag consumati in un buffer, dal quale il programma agirà sui dati.

Espressioni matematiche

In questa sezione viene illustrato in che modo un programmatore di S7 può eseguire calcoli matematici in Logix. Come esempio sarà utilizzata l'espressione “v(cos(x)^2 + sin(x)^2)”. Il risultato di questa espressione è sempre esattamente 1, pertanto è facile verificare se si ottiene la risposta corretta.

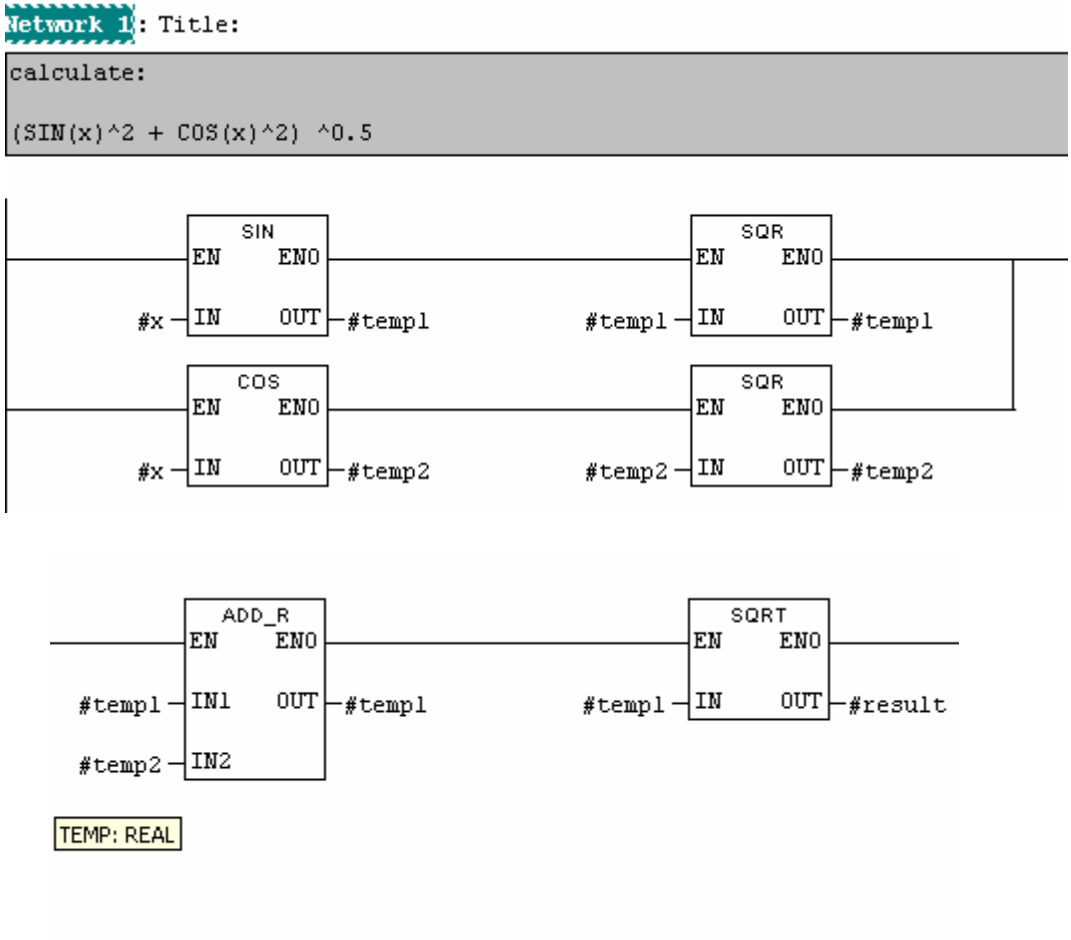
STEP 7 - STL

Il codice matematico in STL di STEP 7 è efficace, ma forse non troppo chiaro per coloro che non hanno esperienza con STL.

```
Network 1: Title:
calculate:

(SIN(x)^2 + COS(x)^2) ^0.5

          L     #x
          SIN
          SQR
          L     #x
          COS
          SQR
          +R
          SQRT
          T     #result
```

STEP 7 - LAD

La valutazione matematica in LAD segue uno schema tradizionale di combinazione delle funzioni.

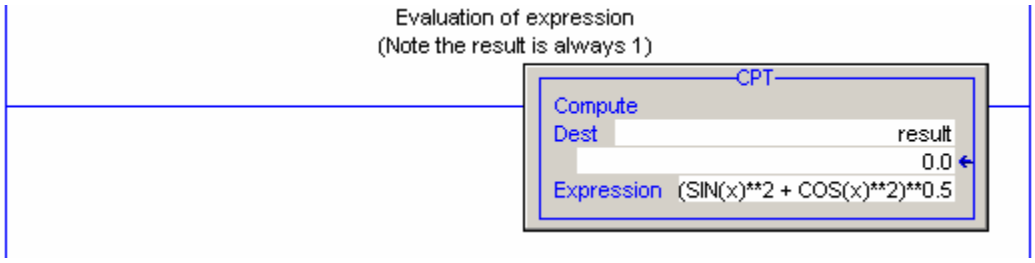


Logix - ST

L'espressione viene immessa allo stesso modo di qualsiasi altro linguaggio di alto livello.

```
// evaluation of mathematical expression in Structured Text

result := (SIN(x)**2 + COS(x)**2)**0.5;
```

Logix - LD



L'istruzione CPT consente di immettere l'espressione in un modo ad alto livello più facilmente comprensibile dalla maggior parte delle persone rispetto a una rete (ramo) di istruzioni separate.

STEP 7 - Funzione utente

Questo blocco funzione è stato scritto per eseguire praticamente le stesse operazioni di CPT di Logix.

```
CALL  "CPT" , "Data_CPT"                    FB119 / DB119
 str   :="test_formulae".test_string15      P#DB16.DBX476.0
 x     :=3.300000e+001
 a     :=0.000000e+000
 b     :=0.000000e+000
 c     :=0.000000e+000
 result:=#realVar
```

```
-- ((cos x)^2 + (sin x)^2) ^0.5 --
```

IN	OUT
+3.300e+001 +0.000e+000 +0.000e+000 +0.000e+000	+1.000e+000

Legge e valuta una stringa di espressione memorizzata in un blocco dati. Ha una limitazione rispetto a CPT di Logix: l'espressione è scritta in notazione polacca inversa, non compressibile da tutti.

Il problema principale nella scrittura di un blocco funzione come questo è che richiede tempo e non è adatto ai programmatori principianti. Con Logix, l'istruzione CPT può essere utilizzata da tutti i tipi di utenti quando viene installato il software RSLogix 5000.

Controllo del tipo

Con STEP 7 e Logix, i parametri a funzioni, blocchi funzione, istruzioni e istruzioni add-on sono rigidamente controllati per tipo dai compilatori.

Esistono differenze con le espressioni matematiche.

Logix fa distinzione tra valori numerici e booleani. I compilatori rifiuteranno espressioni che uniscono in modo illogico valori numerici e booleani. Quando rileva espressioni con tipo numerico misto, eseguirà conversioni per generare un risultato del tipo di variabile di risultato dichiarata. Pertanto interpreterà * come moltiplicazione di numeri interi se il risultato deve essere un numero intero e come moltiplicazione di numeri reali se il risultato deve essere un numero reale.

In STEP 7, è necessario specificare il tipo di operazioni aritmetiche. Sono presenti, ad esempio, *I (moltiplicazione di due numeri interi a 16 bit) *D (moltiplicazione di due numeri interi a 32 bit) e *R (moltiplicazione di due numeri). Il programmatore deve assicurarsi che i due numeri operandi di un'istruzione *R siano reali. In caso contrario, il compilatore non rileverà il problema, ma il risultato non avrà senso.

Conclusione

I metodi di programmazione di Logix delle espressioni matematiche sono più chiari, e separando il codice matematico dall'altra logica, semplificheranno le operazioni di test e convalida.

Altri argomenti relativi alla programmazione

Ambito delle variabili

È un'area in cui Logix differisce da STEP 7.

Regole per STEP 7

- Le variabili temporanee sono invisibili al di fuori del blocco in cui sono dichiarate.
- Le variabili statiche globali sono visibili in tutto il programma.
- Le variabili statiche dichiarate come dati di istanza in un blocco funzione hanno uno stato speciale in FB, ma non è possibile accedervi da altre parti del programma.

Regole di Logix

L'esecuzione in Logix è suddivisa in task. Ciascun task può includere più programmi e ciascun programma diverse routine. Ciascun programma può disporre della propria sezione di tag.

- I tag dell'ambito controllore sono visibili in tutte le routine e in tutti i programmi.
- I tag nell'ambito programma sono visibili solo nelle routine e nel programma nel quale vengono definiti. Ciò significa che se una routine in un programma deve condividere i dati con una routine in un altro programma, deve utilizzare i dati nell'ambito controllore.
- I tag locali dell'istruzione add-on sono visibili solo alla relativa logica dell'istruzione add-on.

OB, task e schedulazione

Blocchi organizzativi, task e schedulazione sono descritti nel Capitolo 2.

Un esempio più ampio - Modulo di controllo

In questo esempio sono riuniti alcuni degli argomenti illustrati nelle sezioni precedenti. Il termine “Modulo di controllo” (Control Module, CM) deriva dall'autorevole standard di controllo batch S88. S88 ha portato la struttura del software del controllore ad essere più “orientata all'oggetto”. Questo modulo di controllo è per una valvola binaria. L'istruzione add-on è adatta a questo tipo di programmazione.

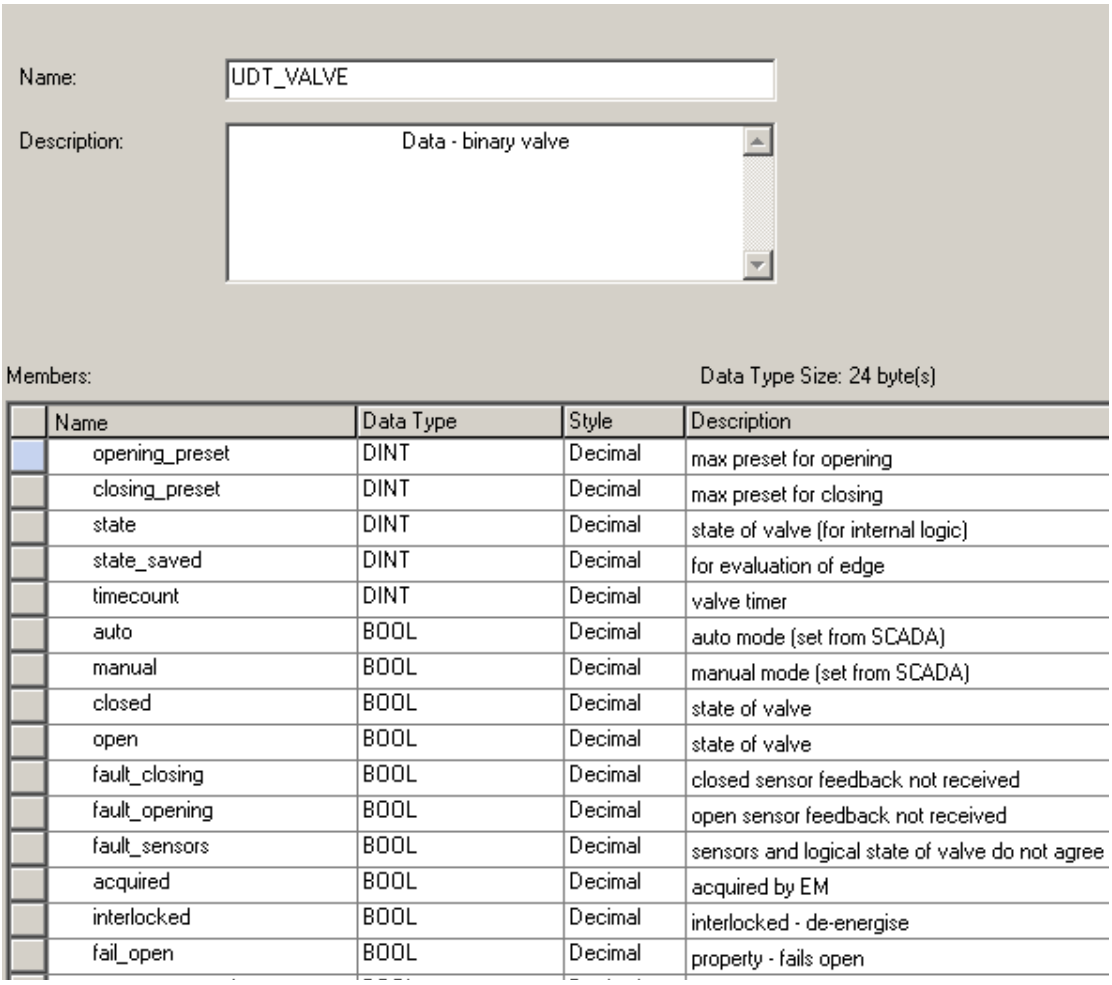
Componenti di CM

Sono inclusi:

- un UDT chiamato UDT_VALVE.
- un'istruzione add-on chiamata AOI_VALVE_2SENSOR
- un nuovo programma in “task_02s” chiamato “valves_callup”, che contiene la sezione dei tag di programma e una routine.

Valvola con tipo di dati utente

L'UDT è mostrato di seguito.



Name: UDT_VALVE

Description: Data - binary valve

Members: Data Type Size: 24 byte(s)

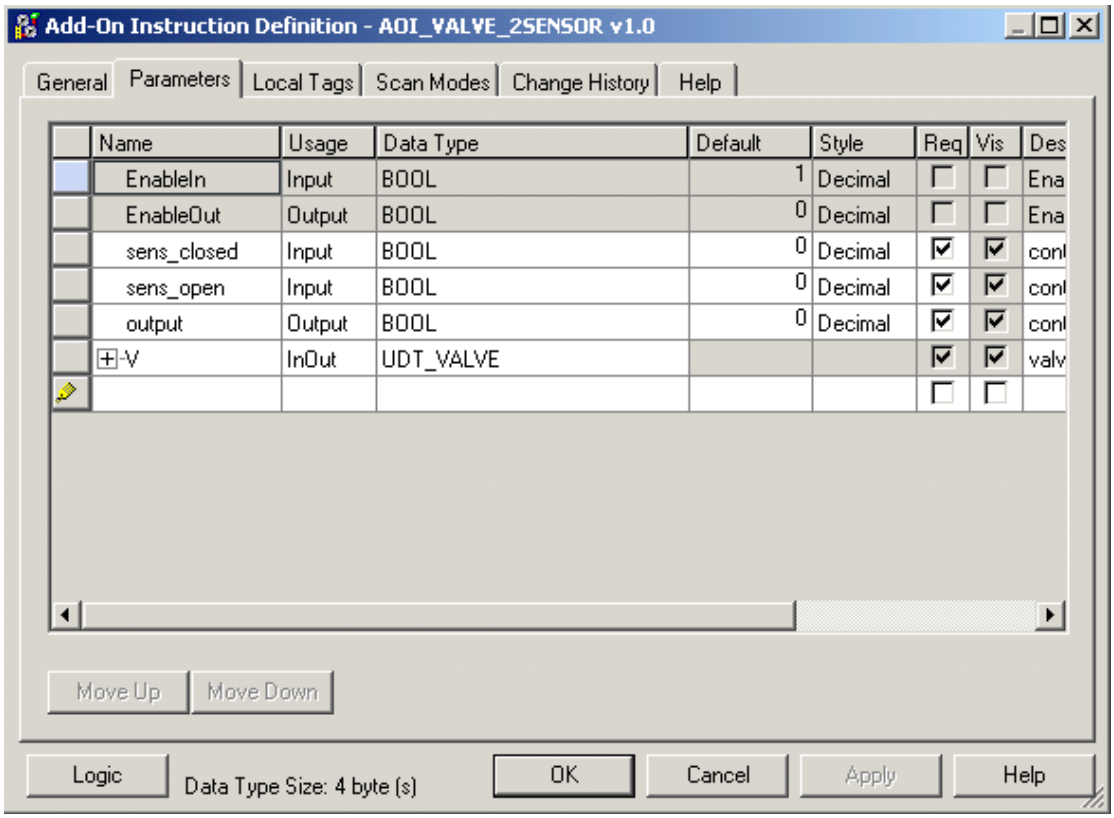
Name	Data Type	Style	Description
opening_preset	DINT	Decimal	max preset for opening
closing_preset	DINT	Decimal	max preset for closing
state	DINT	Decimal	state of valve (for internal logic)
state_saved	DINT	Decimal	for evaluation of edge
timecount	DINT	Decimal	valve timer
auto	BOOL	Decimal	auto mode (set from SCADA)
manual	BOOL	Decimal	manual mode (set from SCADA)
closed	BOOL	Decimal	state of valve
open	BOOL	Decimal	state of valve
fault_closing	BOOL	Decimal	closed sensor feedback not received
fault_opening	BOOL	Decimal	open sensor feedback not received
fault_sensors	BOOL	Decimal	sensors and logical state of valve do not agree
acquired	BOOL	Decimal	acquired by EM
interlocked	BOOL	Decimal	interlocked - de-energise
fail_open	BOOL	Decimal	property - fails open

La creazione dell'UDT deve essere il primo passaggio: include tutti i dati necessari per modellizzare la valvola.

Istruzione add-on

Parametri dell'istruzione add-on

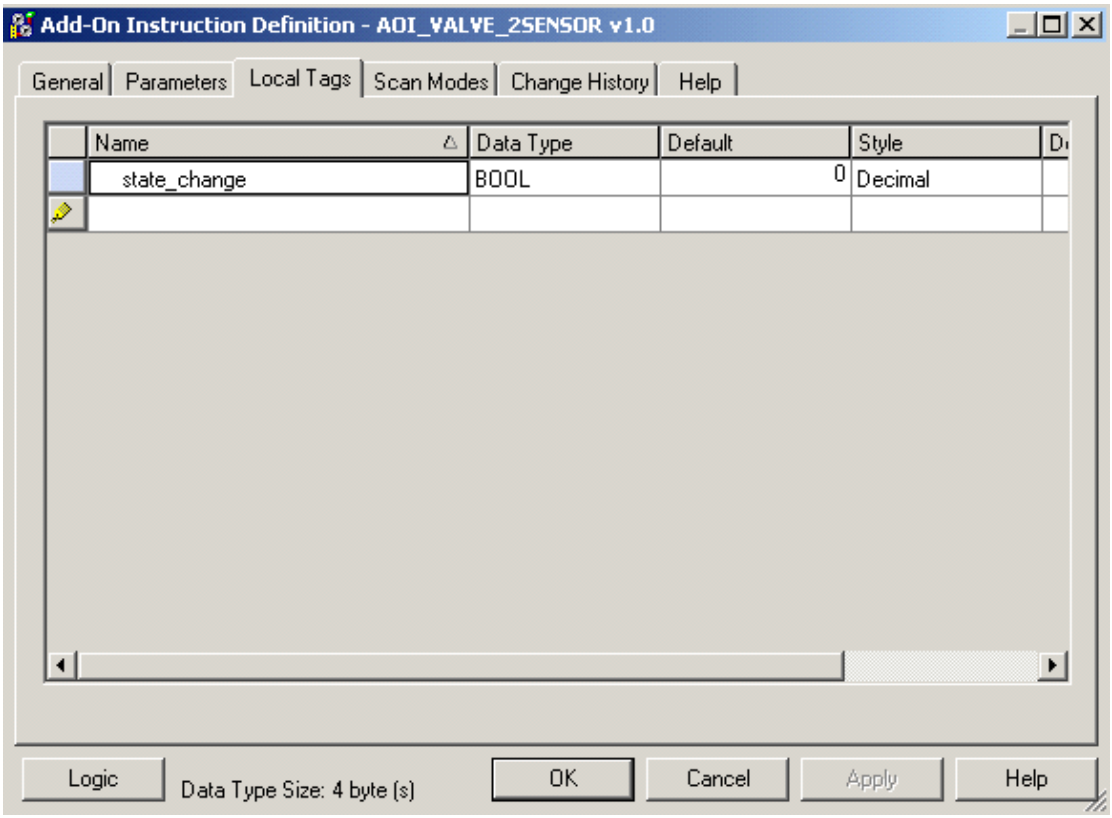
Nell'immagine è illustrata la schermata di configurazione dei parametri.



I parametri aggiunti sono l'I/O per la valvola e un oggetto di tipo "UDT_VALVE". "V" deve essere un parametro InOut.

Dati locali dell'istruzione add-on

Nell'immagine seguente è illustrata la pagina di configurazione per i dati locali dell'istruzione add-on.



Logica dell'istruzione add-on

Nell'immagine seguente è illustrata la logica per questa istruzione add-on.

```
// Control Module Valve_2sensor
// ----------------------------
// Implements logic for a valve with an open and a closed sensor and one output

// See UDT Valve for data structure.

// Note the open/close command V.open_command must be set or reset externally
// and then left until the next activation is required. Do not continuously
// hold the flag set or reset.

// increment timer counter
V.timecount := V.timecount + 1;

// evaluate change of state (state machine)
state_change := V.state <> V.state_saved;
V.state_saved := V.state;

// set output
output := (V.fail_open xor V.open_command) and not
                                        (V.interlocked or V.faulted);

// valve is faulted
V.faulted := V.fault_opening or V.fault_closing or V.fault_sensors;

// action on fault or interlock

if V.faulted or V.interlocked then
    if V.fail_open then
        V.state := 3;
        V.open_command := 1;
    else
        V.state := 0;
        V.open_command := 0;
    end_if;
end_if;


// state machine:
// the state machine does not set outputs - it monitors inputs
// to set status and faults.
case V.state of
    // state 0 - valve is closed - wait for open command
    0:  V.closed := 1;
        V.open := 0;
        if (V.open_Command) then
            V.state := 1;
        // fault sensors
        else
            V.fault_sensors := (not sens_closed) or (sens_open);
        end_if;
    // state 1 -
    1:  V.state := 2;
    // state 2 - waiting for open sensor
    2:  if (sens_open & not sens_closed) then
            V.state := 3;
```

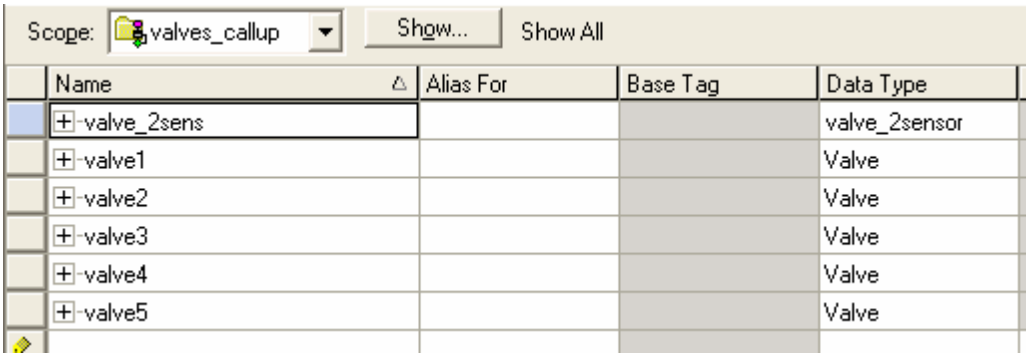
```
        // possible close command while waiting to open
        elsif not V.open_command then
            V.state := 0;
        // fault opening
        else
            V.fault_opening := (V.timecount >  V.opening_preset);
        end_if;
    // state 3 - open - wait for close command
    3:  V.closed := 0;
        V.open := 1;
        if (not V.open_command) then
            V.state := 4;
        // fault sensors
        else
            V.fault_sensors := (sens_closed) or (not sens_open);
        end_if;
    // state 4 -
    4:  V.state := 5;
    // state 5 - wait for closed sensor
    5:  if (sens_closed & not sens_open) then
            V.state := 0;
        // possible open command while waiting to close
        elsif V.open_command then
            V.state := 3;
        // fault closing
        else
            V.fault_closing := (V.timecount >  V.closing_preset);
        end_if;

          _
else;
end_case;
// end state machine

// reset timer if change of state
if (state_change) then V.timecount := 0;
end_if;

// external fault reset
if (V.clear_faults) then
    V.fault_opening := 0;
    V.fault_closing := 0;
    V.fault_sensors := 0;
    V.clear_faults := 0;
end_if;
```

I tag ai quali si fa riferimento in questa logica sono tutti parametri o tag locali. Ciò significa che l'istruzione add-on può essere utilizzata in qualsiasi programma (ammesso che sia presente anche la valvola UDT).

Richiamo

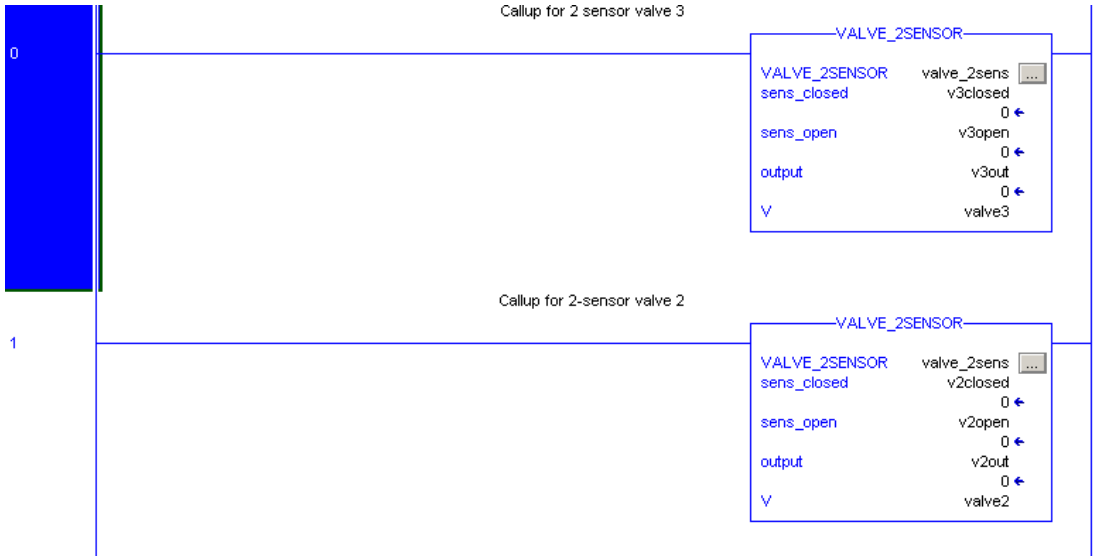
Sia il codice di richiamo che le istanze della valvola UDT si trovano ne programma “valves_callup”, che viene eseguito in task_02s. La frequenza con cui il codice di richiamo viene eseguito varia in base all'applicazione e alla dimensione della valvola.

Nell'immagine seguente sono illustrate le istanze dei dati.



Aggiungere un'istanza del tipo Valve per ciascuna valvola fisica. Il primo tag è il “tag di supporto” richiesto per l'istruzione add-on.

Nell'immagine seguente è illustrato il codice di richiamo.



Chiamare l'istruzione add-on una volta per ciascuna valvola. I parametri effettivi sono i tag I/O correnti per i sensori e il solenoide della valvola e l'istanza della "valvola" UDT.

I tag I/O appariranno solo nella chiamata all'istruzione add-on. Non saranno utilizzati in nessun'altra parte del programma. Oltre a essere più pulito dal punto di vista della struttura del software, elimina qualsiasi rischio di problemi derivanti dall'aggiornamento asincrono dell'I/O.

Si ricordi che con i controllori Logix, gli I/O vengono sottoposti a scansione in modo asincrono.

Errori comuni durante la conversione in Logix

Introduzione

L'obiettivo di questa sezione è evidenziare alcuni errori di progettazione e programmazione spesso commessi dagli utenti di S7 quando convertono le applicazioni in Logix. Tali errori sono stati identificati esaminando i programmi Logix convertiti da STEP 7.

Argomento	Pagina
Se non si seleziona l'hardware appropriato	129
Sottovalutazione dell'impatto della schedulazione dei task	130
Esecuzione della traduzione anziché della conversione	130
Se non si utilizzano i linguaggi appropriati	130
Implementazione di tipi di dati errati – DINT e INT	131
Codice utente che emula istruzioni esistenti	132
Utilizzo errato di COP, MOV e CPS	133
Utilizzo errato di CPT	133
Se le stringhe non sono gestite in modo ottimale	133
Utilizzo estensivo di salti	133
Se non si utilizzano tag alias	133

Gli errori di programmazione rientrano nelle seguenti due categorie:

- programmazione che comporta un'efficienza ridotta del controllore
- programmazione che comporta un sistema di controllo difficile da comprendere, manutenere e sviluppare.

Nella maggior parte dei casi, il codice per migliorare l'efficienza consente di migliorare anche la leggibilità e la modularità del programma. Viceversa, migliorandone la struttura, il programma dovrebbe diventare anche più efficiente.

Se non si seleziona l'hardware appropriato

Questo capitolo è incentrato soprattutto sul software. Si noti, tuttavia, che la selezione corretta dell'hardware è un requisito fondamentale per un funzionamento soddisfacente. È possibile che il numero di controllori e di rack non corrisponda a quello di un sistema S7 equivalente.

Per ulteriori informazioni sull'hardware, consultare il Capitolo 1 e l'Appendice A. Ulteriori informazioni sono reperibili nelle Appendici A e B.

Sottovalutazione dell'impatto della schedulazione dei task

Nell'area di schedulazione e interrupt, non vi è molta differenza tra i due sistemi. Tuttavia, nell'ambiente Logix, la schedulazione è maggiormente incoraggiata.

È abbastanza comune che i programmatori di STEP 7 tralascino la schedulazione quando lavorano con i controllori Logix. Per ulteriori informazioni sulla schedulazione in Logix, fare riferimento al Capitolo 2.

Esecuzione della traduzione anziché della conversione

L'esecuzione della traduzione linea per linea di un programma STEP 7 in Logix è un errore comune.

È necessario invece un processo più approfondito, definito conversione, che copre la scelta dei linguaggi, la schedulazione e la selezione delle routine del codice.

Eseguendo la conversione anziché la traduzione dei programmi STEP 7, sarà possibile utilizzare in modo ottimale la capacità del sistema Logix.

Se non si utilizzano i linguaggi appropriati

I programmatori spesso trascurano i linguaggi Logix diversi dalla logica ladder.

Per ulteriori informazioni su come scegliere un linguaggio Logix, fare riferimento al Capitolo 2 e per esempi del codice STEP 7 tradotto in Logix, fare riferimento al Capitolo 4.

Implementazione di tipi di dati errati – DINT e INT

In genere, è consigliabile utilizzare DINT anziché INT.

Nell'esempio riportato di seguito è mostrata la differenza tra la somma di due DINT e di due INT.

Somma di DINT

```
// add two DINTs and assign to a third DINT

for index := 0 to 999 do
    result_DINT := operandA_DINT + operandB_DINT;
end_for;
```

Somma di INT

```
// add two INTs and assign to a third INT

for index := 0 to 999 do
    result_INT := operandA_INT + operandB_INT;
end_for;
```

Risultati temporali

Nella tabella sono mostrati i tempi relativi (il numero più piccolo indica tempi più rapidi). I numeri riportati servono solo per il confronto con altri numeri nella tabella. Non devono essere confrontati con voci in altre tabelle.

Metodo	Tempi relativi
Somma di DINT con loop ST For	53
Somma di INT con loop ST For	100

A scopo di confronto, la stessa prova è stata eseguita con un controllore S7. In questo caso, i risultati sono identici per DINT e INT.

La lezione è utilizzare DINT per tutto il lavoro con numeri interi in Logix. Utilizzare INT o SINT solo se si interfaccia con un sistema esterno che richiede l'uso di INT o SINT.

Codice utente che emula istruzioni esistenti

I programmatori spesso scrivono codice utente quando un'istruzione esistente è ancora in esecuzione. Come esempio, confrontare la copia di un array con codice utente e con l'istruzione COP.

Codice utente

```
for index := 0 to 99 do
    target_DINT[index] := source_DINT[index];
end_for;
```

Istruzione COP

```
cop(source_DINT[0], target_DINT[0], 100);
```

Di seguito sono riportati i tempi relativi ai due metodi. Anche in questo caso, i numeri riportati servono solo per il confronto con altri numeri nella tabella. Non devono essere confrontati con voci in altre tabelle.

Metodo	Tempo relativo
Copia di array di DINT con testo strutturato	100
Copia di array di DINT con COP	18

Per eseguire operazioni, come la copia di array, sono utilizzate le funzioni della libreria di STEP 7 scritte nell'elenco di istruzioni. Se la funzione della libreria non esegue l'operazione richiesta, è possibile scriverne una nuova. Le funzioni scritte possono essere efficaci quasi quanto quelle fornite da STEP 7.

Tuttavia, in Logix, i programmatori non possono scrivere una funzione di copia efficace quanto la funzione COP incorporata. La lezione per i programmatori di S7 è di consultare attentamente la Guida alle istruzioni nel software RSLogix 5000 prima di procedere autonomamente.

Utilizzo errato di COP, MOV e CPS

MOV copia un valore semplice (immediato o tag) in un tipo di tag semplice: DINT, INT, SINT o REAL. COP è in grado di eseguire la stessa operazione di MOV (l'origine non può essere un valore immediato), tuttavia la sua funzione più importante è la copia di tipi di dati complessi.

L'utilizzo di COP per copiare tipi di dati semplici rappresenta un errore di programmazione secondario.

Un errore commesso spesso è l'utilizzo di più MOV per copiare una struttura di dati quando invece è possibile utilizzare un unico COP.

Se i dati di origine possono variare durante la copia a causa di aggiornamenti di I/O asincroni, utilizzare CPS. Questa istruzione non può essere interrotta, pertanto i dati di origine rimarranno costanti durante la copia.

Utilizzo errato di CPT

In Logix, l'istruzione CPT può essere utilizzata per valutare le espressioni. L'espressione è immessa in uno dei campi dell'istruzione. Questa funzione è molto pratica.

Tuttavia, CPT deve essere utilizzato solo se sono necessarie più istruzioni aritmetiche per valutare l'espressione. Se è sufficiente un'unica istruzione, sarà più veloce di CPT.

Per ulteriori informazioni su CPT, fare riferimento al Capitolo 4.

Se le stringhe non sono gestite in modo ottimale

Se è necessario definire un nuovo tipo di stringa, ad esempio con un numero di caratteri diverso rispetto al numero predefinito 82, la creazione di un nuovo “Tipo di dati utente” sarebbe un errore. Creare invece un nuovo tipo di dati di stringa. Il vantaggio di questo metodo consiste nell'aggiornamento automatico del campo ‘LEN’ man mano che la lunghezza della stringa cambia.

Utilizzo estensivo di salti

In Logix, i salti possono verificarsi solo nella logica ladder. Si consiglia di utilizzare l'istruzione JMP in modo limitato. I salti nella logica ladder spesso rendono il programma di difficile lettura.

Se non si utilizzano tag alias

Si ricordi di creare tag alias per i tag I/O creati dal software RSLogix 5000. Renderanno il programma più facile da leggere. Fare riferimento al Capitolo 2.

Note:

Glossario S7 - Logix

Introduzione

In questo capitolo è fornito un glossario dei termini di S7 e dei relativi equivalenti Logix.

Terminologia hardware

Termine di S7	Definizione	Termine di Logix	Definizione
Processore di comunicazione	Modulo di comunicazione	Ponte (Bridge)	
Controllore	Controllore	Controllore	
CPU	Unità centrale di elaborazione	CPU o controllore	
CPU fail-safe	CPU 315F-2 DP; implementa la versione PROFISAFE di DP	GuardLogix	L61S, L62S, L63S
Industrial Ethernet	Versione Siemens di Ethernet	EtherNet/IP ControlNet	Entrambe queste reti hanno funzionalità uguali (o migliori) a Industrial Ethernet
MPI	Multi-Point Interface – un bus seriale	Seriale	Protocolli DF1 o DH485
Controllore programmabile		Controllore o PAC	
PROFIBUS DP	Bus di campo molto diffuso	EtherNet/IP ControlNet DeviceNet	
PROFIBUS PA	Varietà di Profibus utilizzato nell'automazione di processo	Come Profibus DP	
PROFINET	Profibus su Ethernet	EtherNet/IP	
PROFISAFE	Versione fail-safe di PROFIBUS DP	GuardLogix	
S7-200	Controllori di fascia bassa	MicroLogix	
S7-300	Controllori di fascia media	CompactLogix	
S7-400	Controllori di fascia alta	ControlLogix	
SIMATIC	Marchio dei prodotti di automazione Siemens	Logix	

Terminologia software

Termine di S7	Definizione	Termine di Logix più simile	Definizione
Accumulatore	Utilizzato in STL	N/A	Nei linguaggi Logix non è necessario accedere alle strutture di basso livello della CPU
AR1, AR2	Registri puntatore	N/A	Nei linguaggi Logix non è necessario accedere alle strutture di basso livello della CPU
Array	Sintassi ARRAY[0…7] OF REAL	Array	Sintassi REAL[8] L'indicizzazione inizia sempre da 0
Bit di memoria	Indirizzi M...	N/A	Utilizzo di tag
Trasferimento a blocchi	Copia di blocchi di dati. SFC20 BLK_MOV	COP	Istruzione (utilizzo di MOV per una variabile semplice)
BOOL		BOOL	
BYTE	Parola a 8 bit	SINT	L'utilizzo è sconsigliato (è più lento di DINT), ad eccezione di quando richiesto (ad esempio per i caratteri della stringa)
CFC	Linguaggio opzionale per il controllo di processo	FBD	Linguaggio blocchi funzione standard.
CHAR	Byte come carattere	SINT	
Cycle_Execution	OB1 – Eseguito continuamente	Task continuo	Eseguito continuamente
Blocco dati	Unità di memoria dati statica	Database di tag nell'ambito controllore o database di tag nell'ambito programma	Globale Visibile dall'interno del programma al quale è collegato il database
DINT	Numero intero doppio	DINT	Numero intero doppio
DWORD	Parola a 32 bit	DINT	
FBD	Diagramma a blocchi funzione	FBD	Diagramma a blocchi funzione
Funzione	Unità di programma con memoria temporanea senza memoria statica	Routine AOI (istruzione add-on)	Entrambe **potrebbero** corrispondere a una funzione
Blocco funzione	Unità di programma con memoria temporanea e memoria statica	Routine AOI (istruzione add-on) Programma	Tutte **potrebbero** corrispondere a un blocco funzione
GRAPH	Linguaggio grafico opzionale	Diagramma funzionale sequenziale	Linguaggio grafico standard
HW Config	Configurazione hardware – Componente di STEP 7	Configurazione I/O	Ramificazione dell'organizer del controllore

Termine di S7	Definizione	Termine di Logix più simile	Definizione
INT	Numero intero	INT	L'uso è sconsigliato (è più lento di DINT)
Interrupt_Execution	OB eseguito periodicamente	Task periodico	Task eseguito periodicamente
LAD	Logica Ladder	LD	Logica Ladder
Libreria	Funzioni di sistema	GSV, SSV	Istruzioni – Get System Value Set System Value
NetPro	Configuratore rete	N/A	Parte della diramazione di configurazione I/O dell'organizer controllore.
Blocco organizzativo	Unità di programma chiamata dal sistema operativo	Task	Unità di programma chiamata dal sistema operativo
Puntatore	Puntatore dati utilizzato in STL	N/A	Utilizzo di array
REAL	Numero a virgola mobile a 32 bit	REAL	Numero a virgola mobile a 32 bit
SCL	Linguaggio opzionale di alto livello	Testo strutturato	Linguaggio standard
Simatic Manager	Componente di STEP 7	Organizer controllore	Componente di RSLogix 5000
STEP 7	Software di sviluppo e monitoraggio per S7	RSLogix 5000	Software di sviluppo e monitoraggio per Logix
STL	Lista istruzioni	N/A	Usare testo strutturato, logica ladder o diagramma funzionale sequenziale
STRING	Sequenza di CHAR. Lunghezza predefinita 254	STRING	Sequenza di SINT. Lunghezza predefinita 82. L'oggetto stringa contiene anche la relativa lunghezza come proprietà. LEN
STRUCT	Raccolta di dati atipizzata	N/A	In Logix una struttura è un'istanza del tipo (UDT)
Simbolo	Nome dell'indirizzo di memoria dati	Tag	Il tag definisce la struttura della variabile *e* inverte la memoria
Memoria temporanea	Memoria creata su stack run-time	N/A	Usare i tag
WORD	Parola a 16 bit	INT	
UDT	Tipo di dati utente	UDT	Tipo di dati utente

Note:

Componenti di S7 300 e S7 400 ed equivalenti RA

Introduzione

In questa appendice sono elencati i prodotti Siemens e i relativi equivalenti Rockwell Automation.

Argomento	Pagina
CPU Compact S7 300	140
CPU S7 300 standard	140
CPU Technology S7 300	141
CPU S7 300 fail-safe	142
Moduli di ingresso digitali S7 300	142
Moduli di uscita digitali S7 300	143
Moduli di uscita a relè S7 300	144
Moduli digitali misti S7 300	144
Moduli di ingresso analogici S7 300	144
Moduli di uscita analogici S7 300	145
Moduli analogici misti S7 300	146
Moduli di uscita analogici S7 300	146
Controllori ridondanti e fail-safe	147
Moduli di ingresso digitali	147
Moduli di uscita digitali	147
Moduli di ingresso analogici	148
Moduli di uscita analogici	148

CPU Compact S7 300

Numero di catalogo Siemens	Riferimento breve Siemens	Memoria		Porte di comunicazione		Dimensione MMC max.		I/O incorporato			Soluzione RA
			MPI	DP	Seriale		DI	DO	AI	AO	
6ES7 312-5BE0x-xxxx	S7-312C	32K	S	N	N	4 MB	10	6			1769-L31 + Compact I/O ML1500
6ES7 313-5BF0x-xxxx	S7-313C	64 K Sì No No	S	N	N	8 MB	24	16	4	2	1769-L31 + Compact I/O ML1500
6ES7 313-6BF0x-xxxx	S7-313C- PtP	64 K	S	N	RS422/ 485	8 MB	16	16			1769-L31 + Compact I/O ML1500
6ES7 313-6CF0x-xxxx	S7-313C- DP	64 K	S	S	N	8 MB	16	16			1769-L31 + Compact I/O ML1500
6ES7 314-6BG0x-xxxx	S7-314C- PtP	96 K	S	N	RS422/ 485	8 MB	24	16	4	2	1769-L31 + Compact I/O ML1500
6ES7 314-6CG0x-xxxx	S7-314C- DP	96 K Sì Sì No 8 MB	S	S	N	8 MB	24	16	4	2	1769-L31 + Compact I/O ML1500

CPU S7 300 standard

Numero di catalogo Siemens	Riferimento breve Siemens	Memoria		Porte di comunicazione		Dimensione memoria carico max. (RAM)	Soluzione RA
			MPI	DP	PN		
6ES7 312-1AE1x-xxxx	S7-312	32 K	S	N	N	4 MB	1769-L31
6ES7 314-1AG1x-xxxx	S7-314	96 K	S	N	N	8 MB	1769-L31
6ES7 315-2AG1x-xxxx	S7-315-2 DP	128 K	S	S	N	8 MB	1769-L3xE o 1769-L3xC
6ES7 315-2EH1x-xxxx	S7-315-2 PN/DP	256 K	S	S	S	8 MB	1769-L3xE o 1769-L3xC

Numero di catalogo Siemens	Riferimento breve Siemens	Memoria		Porte di comunicazione		Dimensione memoria carico max. (RAM)	Soluzione RA
			MPI	DP	PN		
6ES7 317-2AJ1x-xxxx	S7-317-2 DP	512 K	S	S	N	8 MB	1769-L3xE o 1769-L3xC
6ES7 317-2EK1x-xxxx	S7-317-2 PN/DP	1 MB	S	S	S	8 MB	1769-L3xE o 1769-L3xC
6ES7 319-3ELOx-xxxx	S7-319-3 PN/DP	1,4 MB	S	S	S	8 MB	1769-L3xE o 1769-L3xC

CPU Technology S7 300

Numero di catalogo Siemens	Riferimento breve Siemens	Memoria		Porte di comunicazione		Dimensione memoria carico max. (RAM)	Soluzione RA
			MPI	DP	PN		
6ES7 315-6TG1x-xxxx	S7-315T-2 DP	128 K	S	S	S	4 o 8 MB	1768-L43
6ES7 317-6TJ1x-xxxx	S7-317T-2 DP	512 K	S	S	S	4 o 8 MB	1768-L43

CPU S7 300 fail-safe

Numero di catalogo Siemens	Riferimento breve Siemens	Memoria		Porte di comunicazione		Dimensione memoria carico max. (RAM)	Soluzione ControlLogix RA
			MPI	DP	PN		
6ES7 315-6FF1x-xxxx	S7-315F-2 DP	192 K	S	S	N	8 MB	GuardLogix o SmartGuard 600
6ES7 315-2FH1x-xxxx	S7-315F-2 PN/DP	256 K	S	S	S	8 MB	GuardLogix o SmartGuard 600
6ES7 317-6FF0x-xxxx	S7-317F-2 DP	1 MB	S	S	N	8 MB	GuardLogix o SmartGuard 600
6ES7 317-2FK1x-xxxx	S7-317F-2 PN/DP	1 MB	S	S	S	8 MB	GuardLogix o SmartGuard 600

Moduli di ingresso digitali S7 300

Numero di catalogo Siemens	Connettore anteriore	Punti	Tensione	Soluzione RA	Commenti
6ES7 321-1BH0x-xxxx	20 pin	16	24 V CC	1769-IQ16 1769-IQ16F	
6ES7 321-1BH5x-xxxx	20 pin	16	24 V CC	1769-IQ16 1769-IQ16F	
6ES7 321-1BL0x-xxxx	40 pin	32	24 V CC	1769-IQ32 1769-IQ32T	
6ES7 321-1CH0x-xxxx	40 pin	16	24 … 48 V	N/D	
6ES7 321-1CH2x-xxxx	20 pin	16	48 … 125 V CC	N/D	
6ES7 321-1BH1x-xxxx	20 pin	16	24 V CC	1769-IQ16 1769-IQ16F	
6ES7 321-7BH0x-xxxx	20 pin	16	24 V CC	1769-IQ16 1769-IQ16F	
6ES7 321-1FH0x-xxxx	20 pin	16	120 … 230 V CA	1769-IA16	1769-IA16 supporta solo 120 V CA
6ES7 321-1FF0x-xxxx	20 pin	8	120 … 230 V CA	1769-IM12	1769-IM12 supporta solo 230 V CA

6ES7 321-1FF1x-xxxx	40 pin	8	120 … 230 V CA	1769-IA8I	1769-IA8I supporta solo 120 V CA
6ES7 321-1EL0x-xxxx	40 pin	32	120 V CA	N/D	
N/D		16	TTL 5 V CC	1769-IG16	

Moduli di uscita digitali S7 300

Numero di catalogo Siemens	Connettore frontale	Punti	Tensione	Corrente di uscita	Soluzione RA	Commenti
6ES7 332-1FH0x-xxxx	20 pin	16	120/230 V CA	0,5 A	1769-OA16	
6ES7 332-1FF0x-xxxx	20 pin	8	120/230 V CA	2 A	1769-OA8	S7-300 include un fusibile per gruppo
6ES7 332-5FF0x-xxxx	40 pin	8	120/230 V CA	2 A	1769-OA8	S7-300 è fornito in gruppi di 1
6ES7 322-1BH0x-xxxx	20 pin	16	24 V CC	0,5 A	1769-OB16 1769-OB16P	
6ES7 322-1BH1x-xxxx	20 pin	16	24 V CC	0,5 A	N/A	Alta velocità
6ES7 322-1BL0x-xxxx	40 pin	32	24 V CC	0,5 A	1769-OB32 1769-OB32T	
6ES7 322-1BF0x-xxxx	20 pin	8	24 V CC	2 A	1769-OB8	
6ES7 322-8BF0x-xxxx	20 pin	8	24 V CC	0,5 A	1769-OB8	
6ES7 332-1FL0x-xxxx	2x20 pin	32	120 V CA	1 A	N/D	
6ES7 332-5GH0x-xxxx	40 pin	16	24/48 V	0,5 A	N/D	
6ES7 332-1CF0x-xxxx	20 pin	8	48 … 125 V CC		N/D	
N/A		16	TTL 5 V CC		1769-OG16	
N/A		16	24 V CC		1769-OV16	
N/A		32	24 V CC		1769-OV32T	
N/A		16	24 V CC		1769-OB16P	

Moduli di uscita a relè S7 300

Numero di catalogo Siemens	Connettore frontale	Punti	Corrente di uscita	Soluzione RA	Commenti
6ES7 322-1HH0x-xxxx	20 pin	16	2 A	1769-OW16	
6ES7 322-1HF0x-xxxx	20 pin	8	5 A	1769-OW8	
6ES7 322-1HF1x-xxxx	40 pin	8	5 A	1769-OW8I	
6ES7 322-5HF0x-xxxx	40 pin	8	8 A	1769-OW8I	Il modulo S7-300 è fornito con filtro RC e protezione da sovratensione

Moduli digitali misti S7 300

Numero di catalogo Siemens	Connettore anteriore	Punti	Ingressi	Corrente di uscita	Soluzione RA	Commenti
6ES7 323-1BH0x-xxxx	20 pin	8 / 8	24 V CC	24 V CC / 0,5 A	1769-IQ6XOW4	Compact I/O dispone di un numero inferiore di I/O, uscite a relè
6ES7 323-1BL0x-xxxx	40 pin	16 / 16	24 V CC	24 V CC / 0,5 A	N/A	
6ES7 327-1BH0x-xxxx	20 pin	8 / 8	24 V CC	24 V CC / 0,5 A	N/A	8 ingressi; 8 ingressi o uscite (configurabili)

Moduli di ingresso analogici S7 300

Numero di catalogo Siemens	**Connettore anteriore**	**Punti**	**Risoluzione (bit)**	**Tipo**	**Soluzione Compact I/O**	**Commenti**
6ES7 331-1KF0x-xxxx	40	8	13	Tensione, corrente, resistenza Temperatura	1769sc-IF8U 1769-IF8U	
6ES7 331-7KF0x-xxxx	20	8	9 / 12 / 14	Tensione, corrente, resistenza Temperatura	1769sc-IF8U 1769-IF8U	

6ES7 331-7KB0x-xxxx	20	2	9 / 12 / 14	Tensione, corrente, resistenza Temperatura	1769sc-IF8U 1769-IF4	
6ES7 331-7NF0x-xxxx	40	8	16	Tensione Tensione	1769-IF8	
6ES7 331-7NF1x-xxxx	40	8	16	Tensione Tensione	1769-IF8	Include interrupt hardware alla fine del ciclo a differenza di 6ES7 331-7NF0x-xxxx
6ES7 331-7HF0x-xxxx	20	8	14	Tensione Tensione	1769-IF8	
6ES7 331-7PF0x-xxxx	40	8		RTD Resistenza	1769-IR6	
6ES7 331-7PF1x-xxxx	40	8		Termocoppia	1769-IT6	
N/A					1769-IF4I	

Moduli di uscita analogici S7 300

Numero di catalogo Siemens	Connettore anteriore	Punti	Risoluzione (bit)	Tipo	Soluzione RA	Commenti
6ES7 332-5HD0x-xxxx	40	4	12	Tensione Corrente	1769-OF4VI 1769-OF4CI	
6ES7 332-7ND0x-xxxx	20	4	16	Tensione Corrente	1769-OF4VI 1769-OF4CI	
6ES7 332-5HB0x-xxxx	20	2	12	Tensione Corrente	1769-OF2	
6ES7 332-5HF0x-xxxx	20	8	12	Tensione Corrente	1769-OF8V 1769-OF8C	

Moduli analogici misti S7 300

Numero di catalogo Siemens	Connettore anteriore	Punti	Risoluzione (bit)	Tipo	Soluzione RA	Commenti
6ES7 334-0KE0x-xxxx	20	4 / 2	12	Tensione Corrente Pt 100		Uscita solo in tensione
6ES7 334-0CE0x-xxxx	20	4 / 2	8	Tensione e corrente (ingressi e uscite)	1769-IF4XOF2	

Controllori S7 400 standard

Numero di catalogo Siemens	Riferimento breve Siemens	Dimensione memoria di lavoro		Porte di comunica-zione		Dimensione memoria carico max. (RAM)	Soluzione ControlLogix RA
			MPI	DP	PN		
6ES7 412-1XF04-0AB0	CPU 412-1	144 KB	S	S	N	64 MB	1756-L61
6ES7 412-2GX04-0AB0	CPU 412-2	256 KB	S	S	N	64 MB	1756-L61
6ES7 414-2GX04-0AB0	CPU 414-2	512 KB	S	S	N	64 MB	1756-L62
6ES7 414-3XJ04-0AB0	CPU 414-3	1,4 MB	S	S	N	64 MB	1756-L63
6ES7 414-3EM05-0AB0	CPU 414-3 PN/DP	2,8 MB	S	S	S	64 MB	1756-L63
6ES7 416-3XK04-0AB0	CPU 416-2	2,8 MB	S	S	N	64 MB	1756-L63
6ES7 416-3XL04-0AB0	CPU 416-3	5,6 MB	S	S	N	64 MB	1756-L64
6ES7 416-3ER05-0AB0	CPU 416-3 PN/DP	11,2 MB	S	S	S	64 MB	1756-L64
6ES7 417-4XL04-0AB0	CPU 417-4	20 MB	S	S	N	64 MB	1756-L64

Controllori ridondanti e fail-safe

Numero di catalogo Siemens	Riferimento breve Siemens	Dimensione memoria di lavoro		Porte di comunicazione			Dimensione memoria carico max. (RAM)	Soluzione ControlLogix RA
			MPI	DP	PN	Porte sincro		
6ES7 414-4HJ04-0AB0	CPU 414-4H	1,4 MB	S	S	N	S	64 MB	1756-L63
6ES7 417-4HL04-0AB0	CPU 417-4H	20 MB	S	S	N	S	64 MB	1756-L64
6ES7 416-2FK04-0AB0	CPU-416F-2	2,6 MB	S	S	N	N	64 MB	1756-L61S

Moduli di ingresso digitali

Numero di catalogo Siemens	Connettore anteriore	Punti	Intervallo	Soluzione RA	Commenti
6ES7 421-7BH01-0AB0 (interrupt/ diagnostica)	48 pin	16	24 V CC	1756-IB16D	
6ES7 421-1BL01-0AA0	48 pin	32	24 V CC	1756-IB32	
6ES7 421-1EL00-0AA0	48 pin	32	120 V CA/CC	1756-IA32	
6ES7 421-1FH20-0AA0	48 pin	16	230 V CA/CC	1756-IM161	
6ES7 421-7DH00 0AB0 (interrupt/ diagnostica)	48 pin	32	24 -60 V CA/CC		

Moduli di uscita digitali

Numero di catalogo Siemens	Connettore frontale	Punti	Tensione	Corrente	Soluzione RA	Commenti
6ES7 422-1FH00-0AA0	48 pin	16	230 V CA	2 A	1756-OA16	
6ES7 422-1HH00-0AA0	48 pin	16	60 V CC 230 V CA (relè)	5 A	1756-OW16I	
6ES7 422 1BH11-0AA0	48 pin	16	24 V CC	2 A	1756-OB16E	
6ES7 422-1BL00-0AA0	48 pin	32	24 V CC	0,5 A	1756-OB32	
6ES7 422-7BL00-0AB0 (Diagnostica)	48 pin	32	24 V CC	0,5 A	1756-OB16D 1756-OB32	

Moduli di ingresso analogici

Numero di catalogo Siemens	Connettore anteriore	Canali	Risoluzione (bit)	Tipo	Soluzione RA	Commenti
6ES7 431-0HH0-0AB0	48 pin	16	13	Tensione Corrente	1756-IF16	16 bit
6ES7 431-1KF00-0AB0	48 pin	8	13	Tensione Corrente Impedenza	1756-IF8	16 bit 4 ingressi differenziali
6ES7 431-1KF10-0AB0	48 pin	8	14-16	Tensione Corrente Termocoppia Termoresistore Impedenza	1756-IR6I 1756-IT6I	6 RTD 6 termocoppie Entrambi 16 bit
6ES7 431-1FK20-0AB0	48 pin	8	14	Tensione Corrente Impedenza	1756-IF16	16 bit
6ES7 431-7QH00-0AB0 (interrupt)	48 pin	16	16	Tensione Corrente Termocoppia Termoresistore Impedenza	1756-IR6I 1756-IT6I	6 RTD 6 termocoppie
6ES7 431-7KF00-0AB0	48 pin	8	16	Tensione Corrente Termocoppia	1756-IT6I	6 canali
6ES7 431-7KF01-0AB0	48 pin	8	16	Termoresistore	1756-IR6I	5 canali

Moduli di uscita analogici

Numero di catalogo Siemens	Connettore anteriore	Canali	Risoluzione (bit)	Tipo	Soluzione RA	Commenti
6ES7 432-1HF00-0AB0	48 pin	8	13	Tensione Corrente	1756-OF8	15 bit

Tabella di riferimento ~~incrociato~~ dei pannelli operatore Siemens

Utilizzare questa appendice per confrontare i pannelli Rockwell Automation con i pannelli Siemens.

Argomento	Pagina
Pannelli SIMATIC Micro ed equivalenti Rockwell Automation	149
Pannelli SIMATIC - serie 7x ed equivalenti Rockwell Automation	151
Pannelli SIMATIC - serie 17x ed equivalenti Rockwell Automation	152
Pannelli SIMATIC - serie 27x ed equivalenti Rockwell Automation	154
Pannelli SIMATIC Multi Panel - serie 27x ed equivalenti Rockwell Automation	156
Pannelli SIMATIC Multi Panel - serie 37x ed equivalenti Rockwell Automation	158

Pannelli SIMATIC Micro ed equivalenti Rockwell Automation

Pannelli SIMATIC Micro					Soluzione Rockwell Automation		
Numero di catalogo Siemens	**Riferimento breve**	**Descrizione**	**Mem.**	**Opzioni di comunicazione**	**Numero di catalogo Rockwell Automation**	**Nome**	**Descrizione**
6AV6640-0BA11-0AX0	SIMATIC OP 73MICRO	Display STN monocromatico da 3 pollici, 160 x 48 pixel, tastierino, solo 24 V CC	128 KB	1xRS485, compatibile con S7-200, nessuna porta stampante	2711P-K4M5D	PanelView Plus 400 tastierino, in scala di grigi	Display STN in scala di grigi 32 livelli da 3,8 pollici, 320 x 240 pixel, comunicazione RS-232, tastierino, 24 V CC, flash da 64 MB, stampa via USB
6AV6545-0AA15-2AX0	SIMATIC TP070 Fuori produzione da aprile 2007	Display STN da 5,7 pollici, modalità blu (4 livelli), 320 x 240 pixel, touchscreen, solo 24 V CC	128 KB	1xRS485, compatibile con S7-200, nessuna porta stampante	2711P-T6M5D	PanelView Plus 600 touchscreen, in scala di grigi	Display STN in scala di grigi 32 livelli da 5,5 pollici, 320 x 240 pixel, comunicazione RS-232, touchscreen, 24 V CC, stampa via USB

Pannelli SIMATIC Micro					**Soluzione Rockwell Automation**		
Numero di catalogo Siemens	**Riferimento breve**	**Descrizione**	**Mem.**	**Opzioni di comunicazione**	**Numero di catalogo Rockwell Automation**	**Nome**	**Descrizione**
6AV6640-0CA01-0AX0	SIMATIC TP 170MICRO Fuori produzione da aprile 2007	Display STN da 5,7 pollici, modalità blu (4 livelli), 320 x 240 pixel, touchscreen, solo 24 V CC, funzionalità applicazione limitata	256 KB	1xRS485, compatibile con S7-200, nessuna porta stampante	2711P-T6M5D	PanelView Plus 600 touchscreen, in scala di grigi	Display STN in scala di grigi 32 livelli da 5,5 pollici, 320 x 240 pixel, comunicazione RS-232, touchscreen, 24 V CC, stampa via USB
6AV6640-0CA11-0AX0	SIMATIC TP 177MICRO	Display STN da 5,7 pollici, modalità blu (4 livelli), 320 x 240 pixel, touchscreen, solo 24 V CC	256 KB	1xRS485, compatibile con S7-200, nessuna porta stampante	2711P-T6M5D	PanelView Plus 600 touchscreen, in scala di grigi	Display STN in scala di grigi 32 livelli da 5,5 pollici, 320 x 240 pixel, comunicazione RS-232, touchscreen, 24 V CC, stampa via USB
6AV6610-0AA01-1CA8	Software WINCC FLEXIBLE MICRO	Software di configurazione e programmazione solo per pannelli Simatic Micro	N/A	N/A	9701-VWSTMENE	Software RSView Studio Machine Edition	Software di configurazione RSView Studio Machine Edition per lo sviluppo e il collaudo di applicazioni HMI a livello macchina

Pannelli SIMATIC - serie 7*x* ed equivalenti Rockwell Automation

Pannelli SIMATIC - serie 7*x*					**Soluzione Rockwell Automation**		
Numero di catalogo Siemens	**Riferimento breve**	**Descrizione**	**Mem.**	**Opzioni di comunicazione**	**Numero di catalogo Rockwell Automation**	**Nome**	**Descrizione**
6AV6641-0AA11-0AX0	SIMATIC OP73	Display STN monocromatico da 3 pollici, 160 x 48 pixel, tastierino, solo 24 V CC	256 KB	1x RS485, S7-200, S7- 300/400 compatibile, nessuna porta stampante	2711P-K4M5D	PanelView Plus 400 tastierino, in scala di grigi	Display STN in scala di grigi 32 livelli da 3,8 pollici, 320 x 240 pixel, comunicazione RS-232, tastierino, 24 V CC, flash da 64 MB, funzionalità di stampa USB
6AV6641-0BA11-0AX0	SIMATIC OP77A	Display STN monocromatico da 4,5 pollici, 160 x 64 pixel, tastierino. Solo 24 V CC	256 KB	1xRS422, 1xRS485, S7-200, S7-300/400, nessuna porta stampante	2711P-K4M5D	PanelView Plus 400 tastierino, in scala di grigi	Display STN in scala di grigi 32 livelli da 3,8 pollici, 320 x 240 pixel, comunicazione RS-232, tastierino, 24 V CC, flash da 64 MB, funzionalità di stampa USB
6AV6641-0CA01-0AX0	SIMATIC OP77B	Display STN monocromatico da 4,5 pollici, 160 x 64 pixel, tastierino, solo 24 V CC	1 MB	1xRS232, 1xRS422, 1xRS485, USB, S7-200, S7-300/400, porta stampante disponibile	2711P-K4M5D	PanelView Plus 400 tastierino, in scala di grigi	Display STN in scala di grigi 32 livelli da 3,8 pollici, 320 x 240 pixel, comunicazione RS-232, tastierino, 24 V CC, flash da 64 MB, funzionalità di stampa USB
6AV6621-0AA01-0AA0	Software WINCC FLEXIBLE COMPACT	Software di configurazione e programmazione per Simatic OP77, OP/TP170, e pannelli Micro	N/A	N/A	9701-VWSTMENE	Software RSView Studio Machine Edition	Software di configurazione RSView Studio Machine Edition per lo sviluppo e il collaudo di applicazioni HMI a livello macchina

Pannelli SIMATIC - serie 17*x* ed equivalenti Rockwell Automation

Pannelli SIMATIC - serie 17*x*					**Soluzione Rockwell Automation**		
Numero di catalogo Siemens	**Riferimento breve**	**Descrizione**	**Mem.**	**Opzioni di comunicazione**	**Numero di catalogo Rockwell Automation**	**Nome**	**Descrizione**
6AV6545-0BA15-2AX0	SIMATIC TP170A modalità blu Fuori produzione da aprile 2007	Display STN da 5,7 pollici, modalità blu (4 livelli), 320 x 240 pixel, touchscreen, solo 24 V CC	320 KB	1xRS232, 1xRS422, 1xRS485, S5, S7-200, S7-300/400 e controllori di terze parti, nessuna porta stampante	2711P-T6M20D	PanelView Plus 600 touchscreen, in scala di grigi	Display STN in scala di grigi 32 livelli da 5,5 pollici, 320 x 240 pixel, comunicazione Ethernet/IP, RS-232, touchscreen, 24 V CC, flash da 64 MB, funzionalità di stampa USB
6AV6545-0BB15-2AX0	SIMATIC TP170B modalità blu Fuori produzione da aprile 2007	Display STN da 5,7 pollici, modalità blu (4 livelli), 320 x 240 pixel, touchscreen, solo 24 V CC	768 KB	2xRS232, 1xRS422, 1xRS485, S5, S7-200, S7-300/400 e controllori di terze parti, porta stampante disponibile	2711P-T6M20D	PanelView Plus 600 touchscreen, in scala di grigi	Display STN in scala di grigi 32 livelli da 5,5 pollici, 320 x 240 pixel, comunicazione Ethernet/IP, RS-232, touchscreen, 24 V CC, flash da 64 MB, funzionalità di stampa USB
6AV6545-0BC15-2AX0	SIMATIC TP170B a colori Fuori produzione da aprile 2007	Display STN a colori (256 colori) da 5,7 pollici, 320 x 240 pixel, touchscreen. Solo 24 V CC	768 KB	2xRS232, 1xRS422, 1xRS485, S5, S7-200, S7-300/400 e controllori di terze parti, porta stampante disponibile	2711P-T6C20D	PanelView Plus 600 touchscreen, a colori	Display TFT a colori da 5,5 pollici, 320 x 240 pixel, profondità colore 18 bit, comunicazione Ethernet/IP, RS-232, touchscreen, 24 V CC, flash da 64 MB, funzionalità di stampa USB
6AV6542-0BB15-2AX0	SIMATIC OP170B modalità blu Fuori produzione da aprile 2007	Display STN da 5,7 pollici, modalità blu (4 livelli), 320 x 240 pixel, tastierino e touchscreen, solo 24 V CC	768 KB	2xRS232, 1xRS422, 1xRS485, S5, S7-200, S7-300/400 e controllori di terze parti, porta stampante disponibile	2711P-B6M20D	PanelView Plus 600 touchscreen e tastierino, in scala di grigi	Display STN in scala di grigi 32 livelli da 5,5 pollici, 320 x 240 pixel, comunicazione Ethernet/IP, RS-232, touchscreen e tastierino, 24 V CC, flash da 64 MB, funzionalità di stampa USB
6AV6642-0DC01-1AX0	SIMATIC OP177B modalità blu	Display STN da 5,7 pollici, modalità blu (4 livelli), 320 x 240 pixel, tastierino e touchscreen, solo 24 V CC	2 MB	1xRS422, 1xRS485, USB, Ethernet, S5, S7-200, S7-300/400, e controllori di terze parti, porta stampante disponibile	2711P-B6M20D	PanelView Plus 600 touchscreen e tastierino, in scala di grigi	Display STN in scala di grigi 32 livelli da 5,5 pollici, 320 x 240 pixel, comunicazione Ethernet/IP, RS-232, touchscreen e tastierino, 24 V CC, flash da 64 MB, funzionalità di stampa USB

Pannelli SIMATIC - serie 17*x*					**Soluzione Rockwell Automation**		
Numero di catalogo Siemens	**Riferimento breve**	**Descrizione**	**Mem.**	**Opzioni di comunicazione**	**Numero di catalogo Rockwell Automation**	**Nome**	**Descrizione**
6AV6642-0AA11-0AX0	SIMATIC TP177A modalità blu	Display STN da 5,7 pollici, modalità blu (4 livelli), 320 x 240 pixel, touchscreen, solo 24 V CC	512 KB	1xRS422, 1xRS485, S7-200, S7-300/400 compatibile, nessuna porta stampante	2711P-T6M20D	PanelView Plus 600 touchscreen, in scala di grigi	Display STN in scala di grigi 32 livelli da 5,5 pollici, display 320 x 240 pixel, comunicazioni EtherNet/IP, RS-232, touchscreen, 24 V CC, flash da 64 MB, funzionalità di stampa USB
6AV6642-0BA01-1AX0	SIMATIC TP177B a colori	Display STN a colori (256 colori) da 5,7 pollici, 320 x 240 pixel, touchscreen. Solo 24 V CC	2 MB	1xRS422, 1xRS485, USB, Ethernet, S5, S7-200, S7-300/400, e controllori di terze parti, porta stampante disponibile	2711P-T6C20D	PanelView Plus 600 touchscreen, a colori	Display TFT a colori da 5,5 pollici; 320 x 240 pixel, profondità colore 18 bit, comunicazioni EtherNet/IP, RS-232, touchscreen, 24 V CC, flash da 64 MB, funzionalità di stampa USB
6AV6642-0BC01-1AX0	SIMATIC TP177B modalità blu	Display STN da 5,7 pollici, modalità blu (4 livelli), 320 x 240 pixel, touchscreen, solo 24 V CC	2 MB	1xRS422, 1xRS485, USB, S5, S7-200, S7-300/400 e controllori di terze parti, porta stampante disponibile	2711P-T6M20D	PanelView Plus 600 touchscreen, in scala di grigi	Display STN in scala di grigi 32 livelli da 5,5 pollici, display 320 x 240 pixel, comunicazioni EtherNet/IP, RS-232, touchscreen, 24 V CC, flash da 64 MB, funzionalità di stampa USB
6AV6642-8BA10-0AA0	SIMATIC TP177B a colori acciaio inossidabile	Display STN a colori (256 colori) da 5,7 pollici, 320 x 240 pixel, touchscreen, solo 24 V CC, cornice in acciaio inossidabile	2 MB	1xRS422, 1xRS485, USB, Ethernet, S5, S7-200, S7-300/400, e controllori di terze parti, porta stampante disponibile	2711P-T6C20D	PanelView Plus 600 touchscreen, a colori	Display TFT a colori da 5,5 pollici, 320 x 240 pixel, profondità colore 18 bit, comunicazione Ethernet/IP, RS-232, touchscreen, 24 V CC, flash da 64 MB, funzionalità di stampa USB
6AV6642-0DA01-1AX0	SIMATIC OP177B a colori	Display STN a colori (256 colori) da 5,7 pollici, 320 x 240 pixel, tastierino e touchscreen, solo 24 V CC	2 MB	1xRS422, 1xRS485, USB, Ethernet, S5, S7-200, S7-300/400, e controllori di terze parti, porta stampante disponibile	2711P-B6C20D	PanelView Plus 600 touchscreen e tastierino, a colori	Display TFT a colori da 5,5 pollici, 320 x 240 pixel, profondità colore 18 bit, comunicazione Ethernet/IP, RS-232, touchscreen e tastierino, 24 V CC, flash da 64 MB, funzionalità di stampa USB
6AV6621-0AA01-0AA0	Software WINCC FLEXIBLE COMPACT	Software di configurazione e programmazione per Simatic OP77, OP/TP170 e pannelli Micro	N/A	N/A	9701-VWSTMENE	Software RSView Studio Machine Edition	Software di configurazione RSView Studio Machine Edition per lo sviluppo e il collaudo di applicazioni HMI a livello macchina

Pannelli SIMATIC - serie 27x ed equivalenti Rockwell Automation

Pannelli SIMATIC - serie 27*x*					**Soluzione Rockwell Automation**		
Numero di catalogo Siemens	**Riferimento breve**	**Descrizione**	**Mem.**	**Opzioni di comunicazione**	**Numero di catalogo Rockwell Automation**	**Nome**	**Descrizione**
6AV6545-0CA10-0AX0	SIMATIC TP270 a colori da 6 pollici Fuori produzione da ottobre 2006	Display STN a colori (256 colori) da 5,7 pollici, 320 x 240 pixel, touchscreen, solo 24 V CC	2 MB	2xRS232, 1xRS422, 1xRS485, USB, S5, S7-200, S7-300/400 e controllori di terze parti, porta stampante disponibile.	2711P-T6C20D	PanelView Plus 600 touchscreen, a colori	Display TFT a colori da 5,5 pollici, 320 x 240 pixel, profondità colore 18 bit, comunicazione Ethernet/IP, RS-232, touchscreen, 24 V CC, flash da 64 MB, funzionalità di stampa USB
6AV6545-0CC10-0AX0	SIMATIC TP270 a colori da 10 pollici Fuori produzione da ottobre 2006	Display STN a colori (256 colori) da 10,4 pollici, 640 x 480 pixel, touchscreen, solo 24 V CC	2 MB	2xRS232, 1xRS422, 1xRS485, USB, S5, S7-200, S7-300/400 e controllori di terze parti, porta stampante disponibile.	2711P-T10C4D1	PanelView Plus 1000 touchscreen, a colori	Display TFT da 10,4 pollici, 640 x 480 pixel, profondità colore 18 bit, Ethernet/IP e RS-232, touchscreen, 24 V CC, flash da 64 MB, funzionalità di stampa USB
6AV6542-0CA10-0AX0	SIMATIC OP270 a colori da 6 pollici Fuori produzione da ottobre 2006	Display STN a colori (256 colori) da 5,7 pollici, 320 x 240 pixel, tastierino, solo 24 V CC	2 MB	2xRS232, 1xRS422, 1xRS485, USB, S5, S7-200, S7-300/400 e controllori di terze parti, porta stampante disponibile.	2711P-K6C20D	PanelView Plus 600 a colori	Display TFT a colori da 5,5 pollici, 320 x 240 pixel, profondità colore 18 bit, Ethernet/IP, RS-232, tastierino, 24 V CC, flash da 64 MB, funzionalità di stampa USB
6AV6542-0CC10-0AX0	SIMATIC OP270 a colori da 10 pollici Fuori produzione da ottobre 2006	Display STN a colori (256 colori) da 10,4 pollici, 640 x 480 pixel, tastierino, solo 24 V CC	2 MB	2xRS232, 1xRS422, 1xRS485, USB, S5, S7-200, S7-300/400 e controllori di terze parti, porta stampante disponibile.	2711P-K10C4D1	PanelView Plus 1000 tastierino, a colori	Display TFT a colori da 10,4 pollici, 640 x 480 pixel, profondità colore 18 bit, Ethernet/IP e RS-232, tastierino, 24 V CC, flash da 64 MB, funzionalità di stampa USB

Pannelli SIMATIC - serie 27*x*					**Soluzione Rockwell Automation**		
Numero di catalogo Siemens	**Riferimento breve**	**Descrizione**	**Mem.**	**Opzioni di comunicazione**	**Numero di catalogo Rockwell Automation**	**Nome**	**Descrizione**
6AV6643-0AA01-1AX0	SIMATIC TP 227 a colori da 6 pollici	Display STN a colori (256 colori) da 5,7 pollici, 320 x 240 pixel, touchscreen, solo 24 V CC	4 MB	1xRS422, 1xRS485, USB, Ethernet: S5, S7-200, S7-300/400 controllori di terze parti, porta stampante disponibile	2711P-T6C20D	Touchscreen PanelView Plus 600 a colori	Display TFT a colori da 5,5 pollici, 320 x 240 pixel, profondità colore 18 bit, comunicazione Ethernet/IP, RS-232, touchscreen, 24 V CC, flash da 64 MB, funzionalità di stampa USB
6AV6643-0BA01-1AX0	SIMATIC OP 227 a colori da 6 pollici	Display STN a colori (256 colori) da 5,7 pollici, 320 x 240 pixel, tastierino, solo 24 V CC	4 MB	1xRS422, 1xRS485, USB, Ethernet, S5, S7-200, S7-300/400, e controllori di terze parti, porta stampante disponibile	2711P-K6C20D	PanelView Plus 600 a colori	Display TFT a colori da 5,5 pollici, 320 x 240 pixel, profondità colore 18 bit, comunicazione Ethernet/IP, RS-232, tastierino, 24 V CC, flash da 64 MB, funzionalità di stampa USB
6AV6622-0BA01-0AA0	Software WINCC FLEXIBLE STANDARD	Software di configurazione e programmazione per Simatic OP/TP/MP270, MP370, OP77, OP/TP170 e pannelli Micro	N/A	N/A	9701-VWSTMENE	Software RSView Studio Machine Edition	Software di configurazione RSView Studio Machine Edition per lo sviluppo e il collaudo di applicazioni HMI a livello macchina

Pannelli SIMATIC Multi Panel - serie 27*x* ed equivalenti Rockwell Automation

Pannelli SIMATIC Multi Panel - serie 27*x*					**Soluzione Rockwell Automation**		
Numero di catalogo Siemens	**Riferimento breve**	**Descrizione**	**Mem.**	**Opzioni di comunicazione**	**Numero di catalogo Rockwell Automation**	**Nome**	**Descrizione**
6AV6542-0AG10-0AX0	SIMATIC MP270B tastierino 10 pollici Fuori produzione da ottobre 2006	Display TFT a colori (64k colori) da 10,4 pollici, 640 x 480 pixel, tastierino, solo 24 V CC	5 MB	2xRS422, 1xRS485, USB, Ethernet, S5, S7-200, S7-300/400, e controllori di terze parti, porta stampante disponibile	2711P-K10C4D1	PanelView Plus 1000 tastierino, a colori	Display TFT da 10,4 pollici, 640 x 480 pixel, profondità colore 18 bit, comunicazione Ethernet/IP e RS-232, tastierino, 24 V CC, flash da 64 MB, funzionalità di stampa USB
6AV6545-0AG10-0AX0	SIMATIC MP270B touchscreen, 10 pollici Fuori produzione da ottobre 2006	Display TFT a colori (64k colori) da 10,4 pollici, 640 x 480 pixel, touchscreen, solo 24 V CC	5 MB	2xRS422, 1xRS485, USB, Ethernet, S5, S7-200, S7-300/400, e controllori di terze parti, porta stampante disponibile	2711P-T10C4D1	PanelView Plus 1000 touchscreen, a colori	Display TFT da 10,4 pollici, 640 x 480 pixel, profondità colore 18 bit, comunicazione Ethernet/IP e RS-232, touchscreen, 24 V CC, flash da 64 MB, funzionalità di stampa USB
6AV6545-0AH10-0AX0	SIMATIC MP270B touchscreen da 6 pollici Fuori produzione da ottobre 2006	Display TFT a colori (64k colori) da 5,7 pollici, 320 x 240 pixel, touchscreen, solo 24 V CC	5 MB	2xRS422, 1xRS485, USB, Ethernet, S5, S7-200, S7-300/400, e controllori di terze parti, porta stampante disponibile	2711P-K6C20D	PanelView Plus 600 a colori	Display TFT a colori da 5,5 pollici, 320 x 240 pixel, profondità colore 18 bit, comunicazione Ethernet/IP, RS-232, tastierino, 24 V CC, flash da 64 MB, funzionalità di stampa USB
6AV6643-0CB01-1AX0	SIMATIC MP 277 touchscreen da 8 pollici	Display TFT a colori (64k colori) da 7,5 pollici, 640 x 480 pixel, touchscreen, solo 24 V CC	6 MB	1xRS422, 1xRS485, 2xUSB, Ethernet, S5, S7-200, S7-300/400 e controllori di terze parti, porta stampante disponibile	2711P-T7C4D1	PanelView Plus 700 touchscreen a colori	Display TFT da 6,5 pollici, 640 x 480 pixel, profondità colore 18 bit, Ethernet/IP e RS-232, touchscreen, 24 V CC, flash da 64 MB, funzionalità di stampa USB

Pannelli SIMATIC Multi Panel - serie 27*x*					**Soluzione Rockwell Automation**		
Numero di catalogo Siemens	**Riferimento breve**	**Descrizione**	**Mem.**	**Opzioni di comunicazione**	**Numero di catalogo Rockwell Automation**	**Nome**	**Descrizione**
6AV6643-0CD01-1AX0	SIMATIC MP 277 touchscreen da 10 pollici	Display TFT a colori (64k colori) da 10,4 pollici, 640 x 480 pixel, touchscreen, solo 24 V CC	6 MB	1xRS422, 1xRS485, 2xUSB, Ethernet: S5, S7-200, S7-300/400 controllori di terze parti, porta stampante disponibile	2711P-T10C4D1	PanelView Plus 1000 touchscreen a colori	Display TFT da 10,4 pollici, 640 x 480 pixel, profondità colore 18 bit, Ethernet/IP e RS-232, touchscreen, 24 V CC, flash da 64 MB, funzionalità di stampa USB
–	SIMATIC MP 277 touchscreen da 10 pollici, acciaio inossidabile	Display TFT a colori (64 k colori) da 10,4 pollici, 640 x 480 pixel, touchscreen, solo 24 V CC, cornice in acciaio inossidabile, IP66	6 MB	1xRS422, 1xRS485, 2xUSB, Ethernet, S5, S7-200, S7-300/400 e controllori di terze parti, porta stampante disponibile	2711P-T10C4D1	PanelView Plus 1000 touchscreen, a colori	Display TFT da 10,4 pollici, 640 x 480 pixel, profondità colore 18 bit, Ethernet/IP e RS-232, touchscreen, 24 V CC, flash da 64 MB, funzionalità di stampa USB
6AV6643-0DB01-1AX0	SIMATIC MP 277 tastierino, 8 pollici	Display TFT a colori (64k colori) da 7,5 pollici, 640 x 480 pixel, tastierino, solo 24 V CC	6 MB	1xRS422, 1xRS485, 2xUSB, Ethernet, S5, S7-200, S7-300/400 e controllori di terze parti, porta stampante disponibile	2711P-K7C4D1	PanelView Plus 700 tastierino, a colori	Display TFT a colori da 6,5 pollici, 640 x 480 pixel, profondità colore 18 bit, Ethernet/IP e RS-232, tastierino, 24 V CC, flash da 64 MB, funzionalità di stampa USB
6AV6643-0DD01-1AX0	SIMATIC MP 277 tastierino, 10 pollici	Display TFT a colori (64k colori) da 10,5 pollici, 640 x 480 pixel, tastierino, solo 24 V CC	6 MB	1xRS422, 1xRS485, 2xUSB, Ethernet, S5, S7-200, S7-300/400 e controllori di terze parti, porta stampante disponibile	2711P-K10C4D1	PanelView Plus 1000 tastierino a colori	Display TFT da 10,4 pollici, 640 x 480 pixel, profondità colore 18 bit, Ethernet/IP e RS-232, tastierino, 24 V CC, flash da 64 MB, funzionalità di stampa USB
6AV6622-0BA01-0AA0	Software WINCC FLEXIBLE STANDARD	Software di configurazione e programmazione per Simatic OP/TP/MP270, MP370, OP77, OP/TP170 e pannelli Micro	N/A	N/A	9701-VWSTMENE	Software RSView Studio Machine Edition	Software di configurazione RSView Studio Machine Edition per lo sviluppo e il collaudo di applicazioni HMI a livello macchina

Pannelli SIMATIC Multi Panel - serie 37*x* ed equivalenti Rockwell Automation

Pannelli SIMATIC Multi Panel - serie 37*x*					**Soluzione Rockwell Automation**		
Numero di catalogo Siemens	**Riferimento breve**	**Descrizione**	**Mem.**	**Opzioni di comunicazione**	**Numero di catalogo Rockwell Automation**	**Nome**	**Descrizione**
6AV6542-0DA10-0AX0	SIMATIC MP370 tastierino, 12 pollici	Display TFT a colori (256 colori) da 12,1 pollici, 800 x 600 pixel, tastierino, solo 24 V CC	12,5 MB	1xTTY, 2xRS232, 1xRS422, 1xRS485, 1xUSB, Ethernet, S5, S7-200, S7-300/400 e controllori di terze parti, porta	2711P-K12C4D1	PanelView Plus 1250 tastierino, a colori	Display TFT da 12,1 pollici, 800 x 600 pixel, colore a 18 bit, Ethernet/IP e RS-232, tastierino, 24 V CC, flash da 64 MB, funzionalità di stampa USB
6AV6545-0DA10-0AX0	SIMATIC MP370 touchscreen, 12 pollici	Display TFT a colori (256 colori) da 12,1 pollici, 800 x 600 pixel, touchscreen, solo 24 V CC	12,5 MB	1xTTY, 2xRS232, 1xRS422, 1xRS485, 1xUSB, Ethernet, S5, S7-200, S7-300/400 e controllori di terze parti, porta	2711P-T12C4D1	PanelView Plus 1250 touchscreen a colori	Display TFT da 12,1 pollici, 800 x 600 pixel, colore a 18 bit, Ethernet/IP e RS-232, touchscreen, 24 V CC, flash da 64 MB, funzionalità di stampa USB
6AV6545-0DB10-0AX0	SIMATIC MP370 touchscreen, 15 pollici	Display TFT a colori (256 colori) da 15,1 pollici, 1024 x 768 pixel, touchscreen, solo 24 V CC	12,5 MB	1xTTY, 2xRS232, 1xRS422, 1xRS485, 1xUSB, Ethernet, S5, S7-200, S7-300/400 e controllori di terze parti, porta	2711P-T15C4D1	PanelView Plus 1500 touchscreen, a colori	Display TFT da 15 pollici, 1024 x 768 pixel, colore a 18 bit, Ethernet/IP e RS-232, touchscreen, 24 V CC, flash da 64 MB, funzionalità di stampa USB
6AV6545-8DB10-0AA0	SIMATIC MP370 touchscreen, 15 pollici, acciaio inossidabile	Display TFT a colori (256 colori) da 15,1 pollici, 1024 x 768 pixel, touchscreen, solo 24 V CC, cornice in acciaio inossidabile, IP66	12,5 MB	1xTTY, 2xRS232, 1xRS422, 1xRS485, 1xUSB, Ethernet, S5, S7-200, S7-300/400 e controllori di terze parti, porta	2711P-T15C4D1	PanelView Plus 1500 touchscreen, a colori	Display TFT da 15 pollici, 1024 x 768 pixel, colore a 18 bit, Ethernet/IP e RS-232, touchscreen, 24 V CC, flash da 64 MB, funzionalità di stampa USB

Pannelli SIMATIC Multi Panel - serie 37*x*					**Soluzione Rockwell Automation**		
Numero di catalogo Siemens	**Riferimento breve**	**Descrizione**	**Mem.**	**Opzioni di comunicazione**	**Numero di catalogo Rockwell Automation**	**Nome**	**Descrizione**
6AV6 644-0AA01-2AX0	SIMATIC MP377 touchscreen, 12,1 pollici	Display TFT a 65.536 colori da 12,1 pollici, 800 x 600 pixel, touchscreen, solo 24 V CC	12,5 MB	1xTTY, 2xRS232, 1xRS422, 1xRS485, 2xUSB, 2xEthernet, S5, S7-200, S7-300/400 e controllori di terze parti, porta	2711P-T12C4D1	PanelView Plus 1250 touchscreen, a colori	Display TFT da 12,1 pollici, 800 x 600 pixel, colore a 18 bit, Ethernet/IP e RS-232, touchscreen, 24 V CC, flash da 64 MB, funzionalità di stampa USB
6AV6 644-0BA01-2AX0	SIMATIC MP377 tastierino, 12,1 pollici	Display TFT a 65.536 colori da 12,1 pollici, 800 x 600 pixel, tastierino, solo 24 V CC	12,5 MB	1xTTY, 2xRS232, 1xRS422, 1xRS485, 2xUSB, 2xEthernet, S5, S7-200, S7-300/400 e controllori di terze parti, porta	2711P-K12C4D1	PanelView Plus 1250 tastierino, a colori	Display TFT da 12,1 pollici, 800 x 600 pixel, colore a 18 bit, Ethernet/IP e RS-232, tastierino, 24 V CC, flash da 64 MB, funzionalità di stampa USB
6AV6 644-0AB01-2AX0	SIMATIC MP377 touchscreen, 15 pollici	Display TFT a 65.536 colori da 15 pollici, 1024 x 768 pixel, touchscreen, solo 24 V CC	12,5 MB	1xTTY, 2xRS232, 1xRS422, 1xRS485, 2xUSB, 2xEthernet, S5, S7-200, S7-300/400 e controllori di terze parti, porta	2711P-T15C4D1	PanelView Plus 1500 touchscreen, a colori	Display TFT da 15 pollici, 1024 x 768 pixel, colore a 18 bit, Ethernet/IP e RS-232, touchscreen, 24 V CC, flash da 64 MB, funzionalità di stampa USB
6AV6 644-0BA01-2AX0	SIMATIC MP377 touchscreen, 19 pollici	Display TFT a 65.536 colori da 19 pollici, 1280 x 1024 pixel, touchscreen, solo 24 V CC	12,5 MB	1xTTY, 2xRS232, 1xRS422, 1xRS485, 2xUSB, 2xEthernet, S5, S7-200, S7-300/400 e controllori di terze parti, porta	2711P-T15C4D1	Touchscreen PanelView Plus 1500 a colori	Display TFT a colori da 15 pollici, 1024 x 768 pixel, colore a 18 bit, Ethernet/IP e RS-232, touchscreen, 24 V CC, flash da 64 MB, funzionalità di stampa USB
6AV6622-0BA01-0AA0	Software WINCC FLEXIBLE STANDARD	Software di configurazione e programmazione per Simatic OP/TP/ MP270, MP370, OP77, OP/TP170 e pannelli Micro	N/A	N/A	9701-VWSTMENE	Software RSView Studio Machine Edition	Software di configurazione RSView Studio Machine Edition per lo sviluppo e il collaudo di applicazioni HMI a livello macchina

Note:

Assistenza Rockwell Automation

Rockwell Automation fornisce informazioni tecniche online per assistere i clienti nell'utilizzo dei propri prodotti. All'indirizzo http://support.rockwellautomation.com è possibile consultare manuali tecnici, FAQ, note tecniche e applicative, scaricare codici campione e link ai service pack del software e utilizzare la funzione personalizzabile MySupport per sfruttare nel migliore dei modi questi strumenti.

Per ottenere ulteriore assistenza tecnica telefonica per l'installazione, la configurazione e la ricerca guasti, sono disponibili i programmi di assistenza TechConnect. Per maggiori informazioni, rivolgersi al distributore o al rappresentante Rockwell Automation di zona, oppure visitare il sito http://support.rockwellautomation.com.

Assistenza per l'installazione

Se si riscontra un problema entro le prime 24 ore dall'installazione, si prega di consultare le informazioni contenute in questo manuale. Per un aiuto iniziale per l'installazione e la messa in funzione del prodotto, è inoltre possibile contattare un numero specifico dell'Assistenza Clienti.

U.S.A.	1.440.646.3434 Lunedì – Venerdì, 8.00 – 17.00 – orario zona EST
Altre nazioni	Per qualsiasi problema di assistenza tecnica, si prega di contattare il rappresentante Rockwell Automation di zona.

Restituzione di prodotti nuovi non funzionanti

Rockwell Automation collauda tutti i prodotti per garantirne la completa funzionalità al momento della spedizione dalla fabbrica. Tuttavia, se il prodotto non funziona e deve essere restituito, procedere come segue.

U.S.A.	Rivolgersi al proprio distributore. Per completare la procedura di restituzione è necessario fornire al distributore il numero di pratica dell'Assistenza Clienti (per ottenerne uno chiamare i recapiti telefonici citati sopra).
Altre nazioni	Per la procedura di restituzione, si prega di contattare il rappresentante Rockwell Automation di zona.

www.rockwellautomation.com

Power, Control and Information Solutions Headquarters

Americhe: Rockwell Automation, 1201 South Second Street, Milwaukee, WI 53204-2496, USA, Tel: +1 414 382 2000, Fax: +1 414 382 4444
Europa/Medio Oriente/Africa: Rockwell Automation, Vorstlaan/Boulevard du Souverain 36, 1170 Bruxelles, Belgio, Tel: +32 2 663 0600, Fax: +32 2 663 0640
Asia: Rockwell Automation, Level 14, Core F, Cyberport 3, 100 Cyberport Road, Hong Kong, Tel: +852 2887 4788, Fax: +852 2508 1846

Italia: Rockwell Automation S.r.l., Via Gallarate 215, 20151 Milano, Tel: +39 02 334471, Fax: +39 02 33447701, www.rockwellautomation.it

Pubblicazione LOGIX-AP008B-IT-P - Giugno 2008
Sostituisce la pubblicazione LOGIX-AP008B-IT-P

Copyright © 2008 Rockwell Automation, Inc. Tutti i diritti riservati. Stampato in U.S.A.